
# Unreal Beginner's Guide

This guide assumes that you have **NEVER** used Unreal Engine (or any game engine).

Our aim is to give you a step-by-step introduction to how you can quickly get started with your first Tempest Rising custom map. This includes every click and action you have to take to create a map, load your map into Tempest Rising, share with your friends, and start playing.

Every **>CLICK<** or other important action that you have to do in the engine, as a first time map maker, will be in all caps, underlined, and with yellow text on a black background.

This way, step-by-step, we'll guide you through the process of creating a very basic simple/flat 1v1 Tempest Rising map with 2 player spawns and 1 Tempest Field in the middle.

## This guide will introduce you to the following Tempest Rising Map Editor concepts

1) A few Unreal Engine basics (to help you get started)
2) How to create a Tempest Rising Map
3) How to create a new level from a basic map size template
4) How to configure **MapInfo** (the details of your Tempest Rising map, like name, mini-map, player spawns, etc.)
5) How to set up **TedPlayerStarts** (e.g. how to add support for additional players / bots)
6) How to place and configure a **TempestField** (the main resource of Tempest Rising)
7) How to test your level using Tempest Rising
8) How to configure **World Settings** (Fog of War behavior, Mini-Map toggle on/off)
9) How to package and play your Map
10) ***Optional***: How to add Fab Marketplace assets to your Map project

Following this guide should teach you the basics needed to start making maps for Tempest Rising. If you want to create more detailed maps, you will still need to learn basic Unreal Engine workflows, like blockouting, setdressing, etc., which this guide does not currently cover.

Further documentation on Tempest Rising Map Editor features will also not be mentioned here.

For this, consult the **Map Editor Manual** provided with the tool.

# Table of Contents

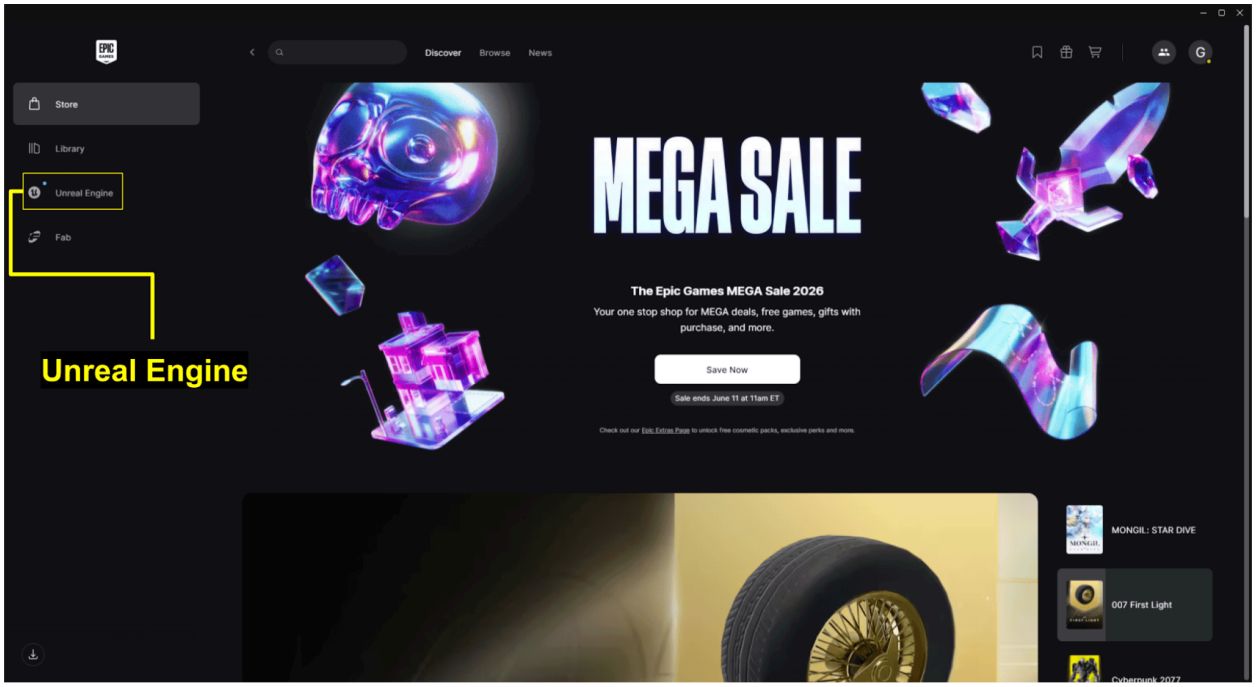
---

# Getting Started!

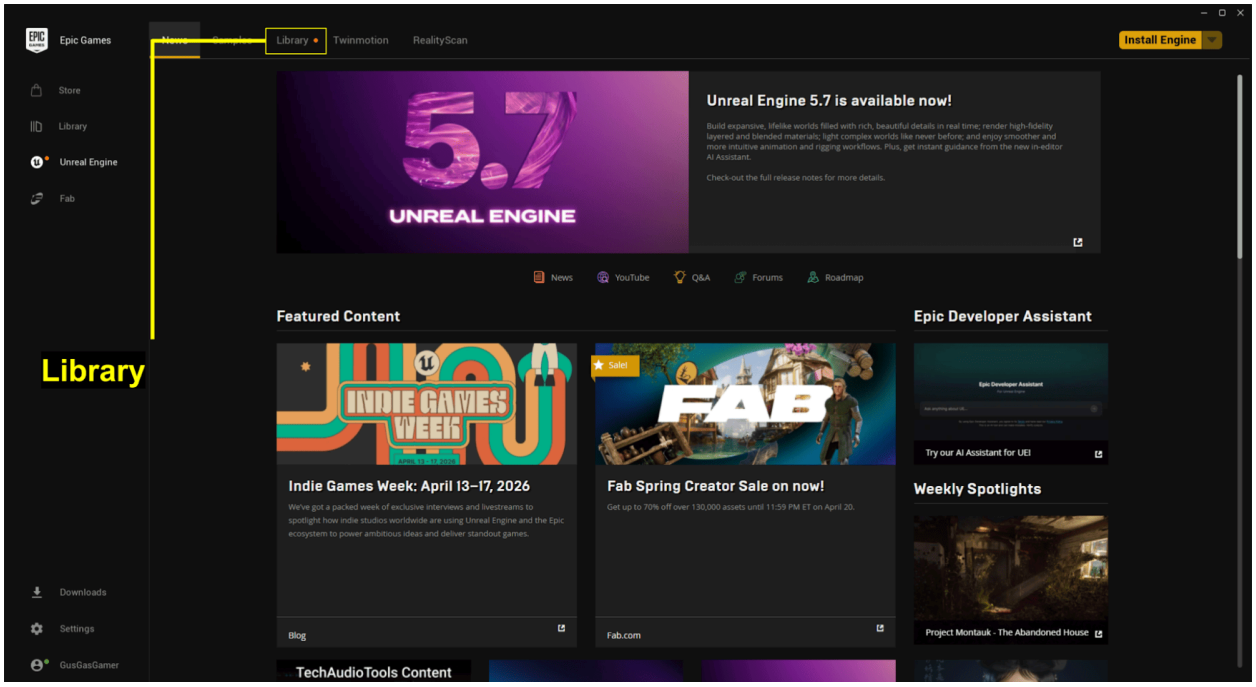
## Installing Unreal Engine

First things first: you need to install **Unreal Engine 5.4.4**. To do this, go to https://store.epicgames.com/download and follow the instructions listed to >DOWNLOAD< the Epic Games Launcher.

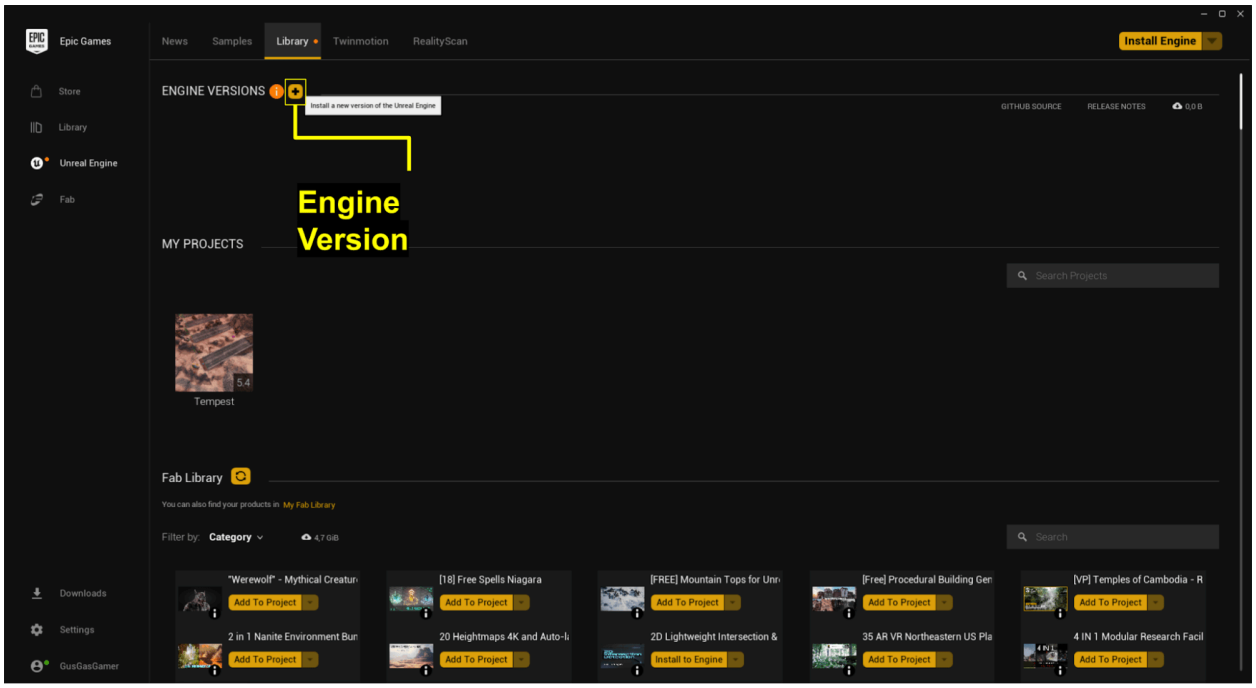
Once you've downloaded the Epic Games Launcher, you will need to **>CLICK<** on **Unreal Engine:**

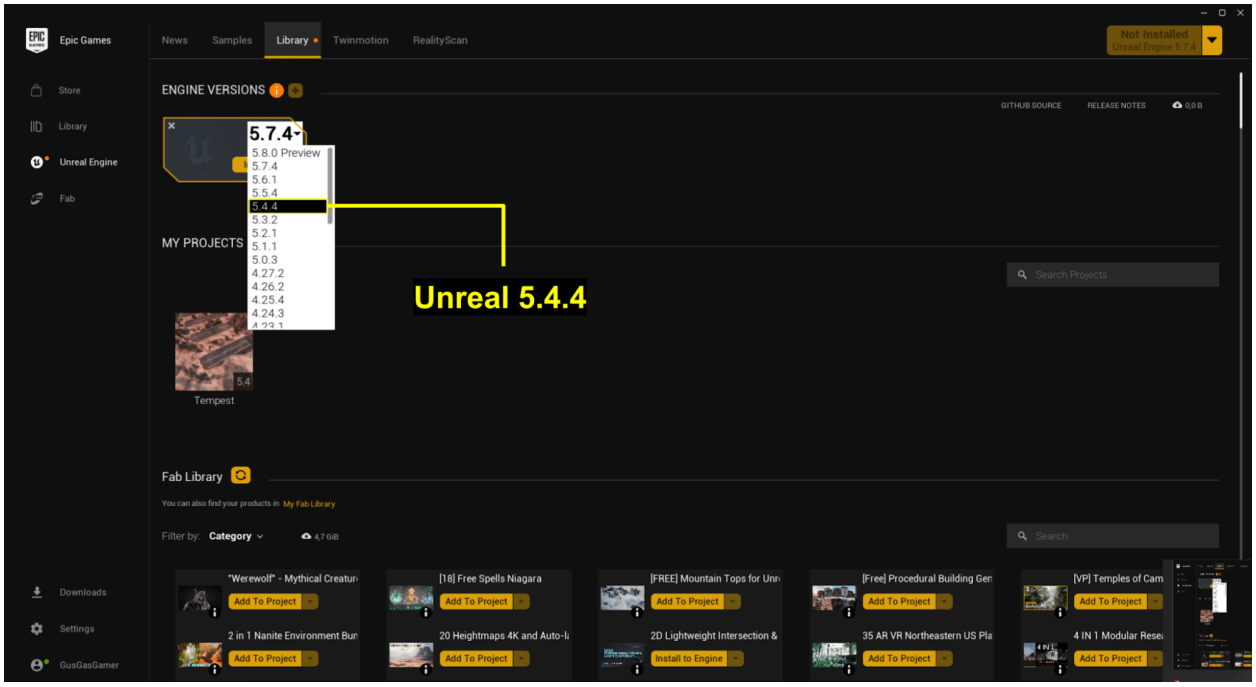



From here, you will need to **>CLICK<** on the **Library** page:




And finally, you will need to **>INSTALL<** **Unreal 5.4.4**:

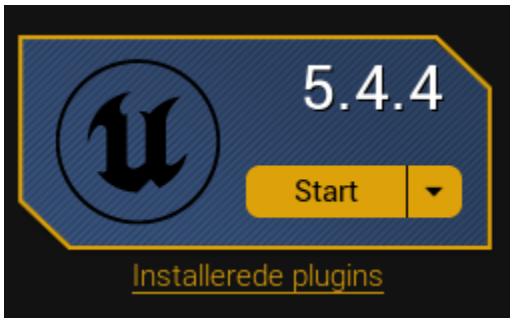




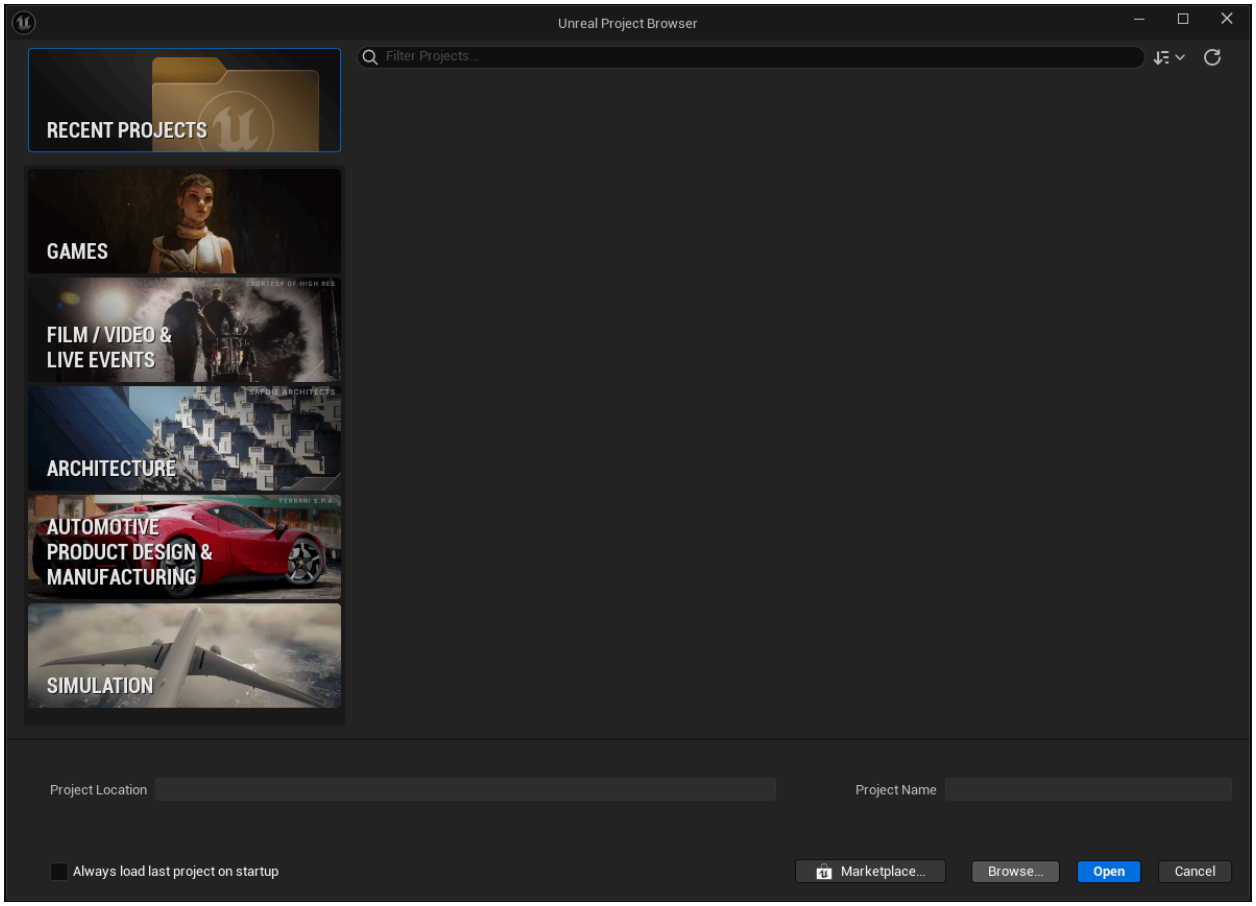

**Unreal Engine version 5.4.4** is the intended Engine Version that is supported by the Tempest Rising Map Editor Project File. However, any version of **5.4.x (x >= 3)** should work.

## Opening the Map Editor Project

Once you've downloaded Unreal Engine 5.4.4, **>CLICK<** Start inside the Epic Games Launcher:



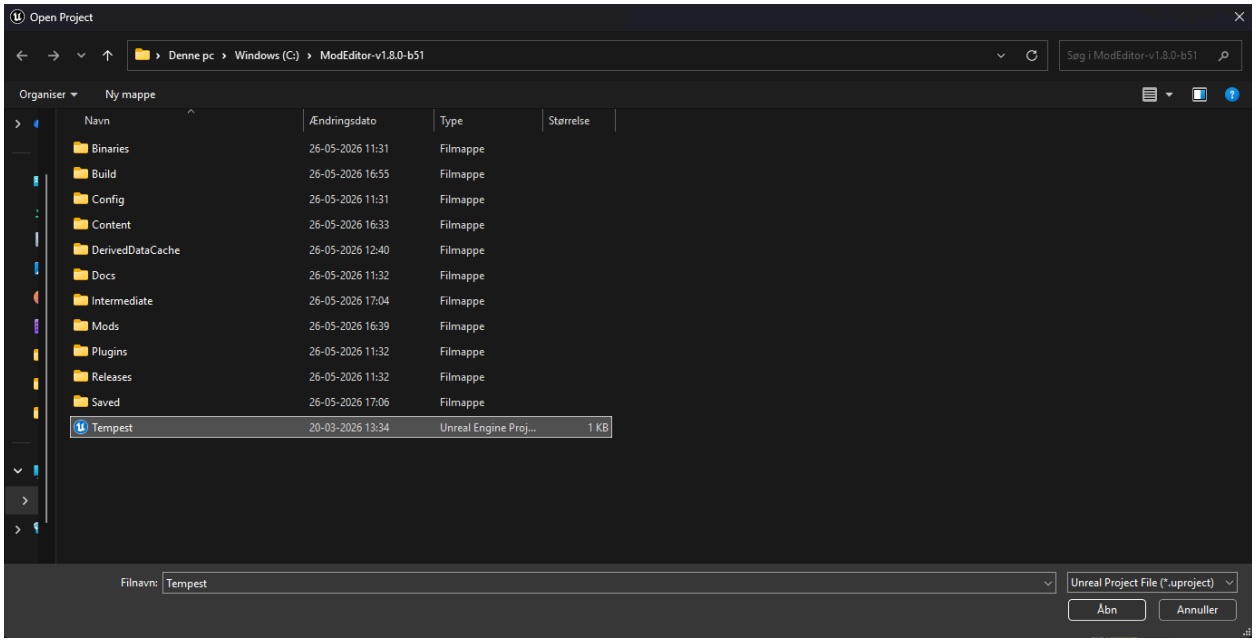

For this next screen, **>CLICK<** 'Browse' in the bottom right corner:




**NOTE**

> *Make sure you have chosen the 'RECENT PROJECTS' panel instead of any of the other options, as otherwise, the 'Browse' option will not be available.*

Navigate to the Map Editor Project (*either inside your Tempest installation folder, or wherever you've placed it*). Select the 'Tempest' Unreal Engine project, and **>CLICK<** "Open" in the bottom right ('*Åbn' because I have a Danish computer, sadly*):




You should now be opening the Map Editor Unreal Engine project!

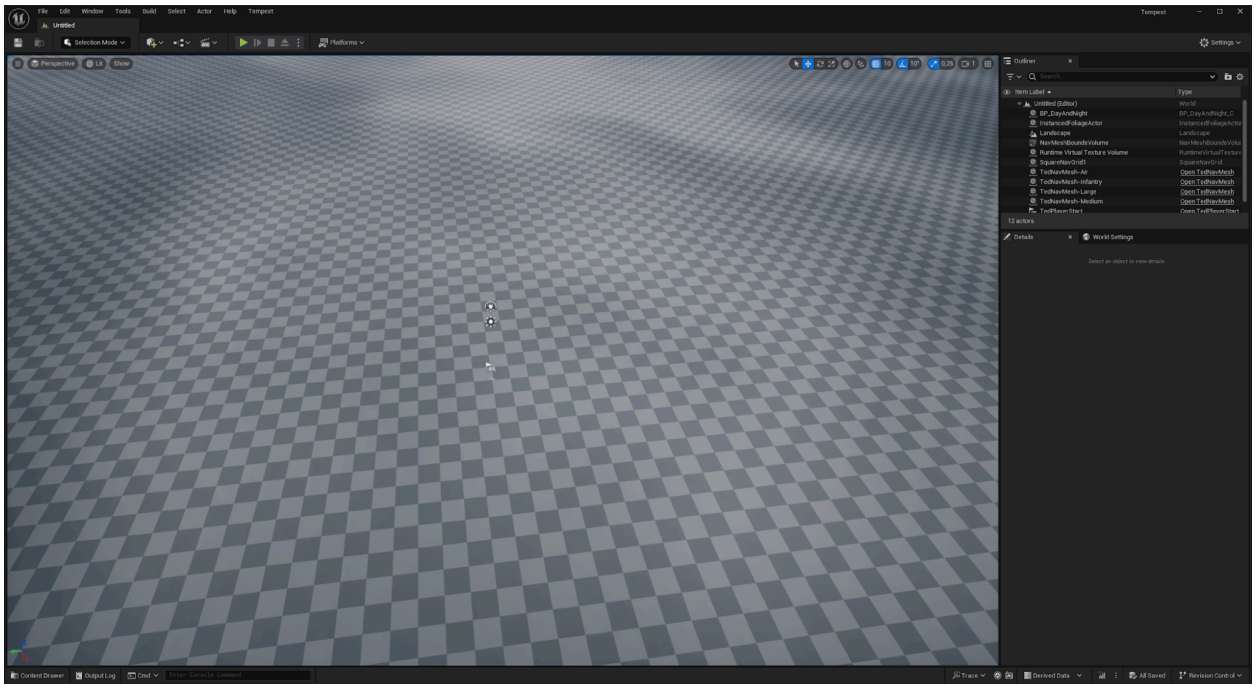
## NOTE

*This may take a few minutes, if it is the first time you are opening the project.*

# Creating Your First Tempest Rising Map

## Unreal Engine Basics

The first thing you'll see when opening the Map Editor Project is the following screen:



This is Unreal Engine 5.4.4's default view, inside an 'Untitled' map. What you see in the background with a grey grid pattern is the default terrain mesh (3D model).

**UNREAL TIP**

*1) If you **hold down the right-mouse button while focused inside the Level Scene, you can use WASD** to move around in the scene.*

*2) If you **hold the right-mouse button and scroll up or down**, you can also change the camera speed at which you fly around in the scene.*

*3) **Left-clicking on an actor, and pressing 'F'**, also zooms you over to / zooms out to see that actor where it is in the scene.*

> ***NOTE:** If you select and press 'F' on the `BP_DayAndNight`, you will be zoomed way out (as this is the 'sun' in the scene). To get back to your starting view, you can select a TedPlayerStart in the Outliner, and press 'F' again.*

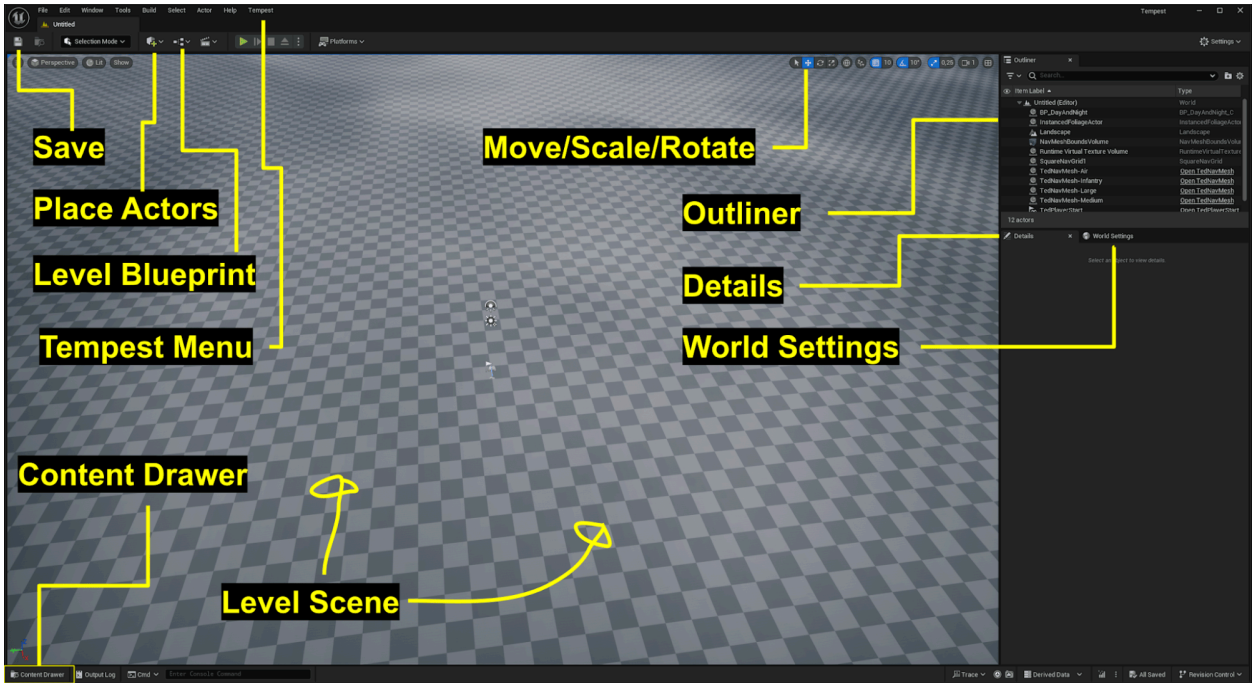
*4) **CTRL + S** quicksaves the file / level you are currently in.*

*5) **CTRL + SHIFT + S** quicksaves every asset you've recently edited.*

*This UE4 Epic Games blog has some nice shortcuts as well!*
*This UE5 video from Epic also has some advanced shortcuts.*

There is a number of cool features / UI elements inside Unreal Engine that we'll quickly go over first, as we'll use these terms often throughout this guide:

For some clarity, here's an overview of where some important things are in the Unreal Engine 5.4.4 view:




- **Level Scene** - meaning the main gameplay scene in which you can place objects.
- **Outliner Panel** - meaning the window in the top right corner where you can see the list of all objects currently inside the Level Scene
- **Content Drawer** - the folder in which all files in the Unreal Project can be found, located at the far bottom left.
- **Top Navigation Bar** - the very top navigation bar. This is where our "Tempest Rising" button lives.
- **Save Button** - This saves the level you are currently in

- **Place Actors Panel** - This gives you fast access to blueprints inside the project, for example TED PlayerStart (which determines where players start when the game begins), or TED TempestField (which you can use to player TempestFields on your map)
- **Move/Scale/Rotate Panel** - This allows you to move, rotate or scale the actor you have selected in the gameplay view.
    - *Note that some actors cannot be scaled (like PlayerStarts, or spawners).*
- **Details Panel** - This shows you all relevant details on the selected actor in the scene. Left-clicking on an actor (for example the terrain) will give you access to its details.
- **World Settings** - This allows you to change certain 'world' details. Note that not all world settings can be changed. Though you can change the Fog of War / Mini-Map behaviour from presets, which we'll be doing later!
- **Level Blueprint** - This is where you can place blueprints / logic that is triggered during gameplay. For simplicity's sake, this Beginner's guide will **NOT** be going into detail on custom logic for Tempest Rising. But know that the Level Blueprint exists, and whatever you put in there will get packaged alongside your map!

**Some other basic Unreal Engine / Game Engine terms:**

- **Actor** - Unreal Engine terminology. Every object inside the gameplay scene is an 'actor'.
- **Blueprint** - Actors that can have behaviour / code inside it. A Tempest Rising Bridge is a blueprint, for example, since it can be destroyed or repaired.
- **Mesh** - a type of 3D model. Typically Unreal Engine uses .fbx files.
- **Asset** - any file you can put into the project; for example a Texture (a 2D image file), a soundfile (.wav), or a Skeletal Mesh (an .fbx 3D model with a rig assigned to it).
- **Node** - A blueprint's 'Event Graph' can contain 'Nodes'. These are pre-built boxes of code that you can use for visual scripting.

**NOTE**

*The Tempest Rising Map Editor project has a few Tempest Rising specific nodes under the `Ted` category; for stuff like spawning units, giving them commands, etc. See the Map Editor Manual for more information.*

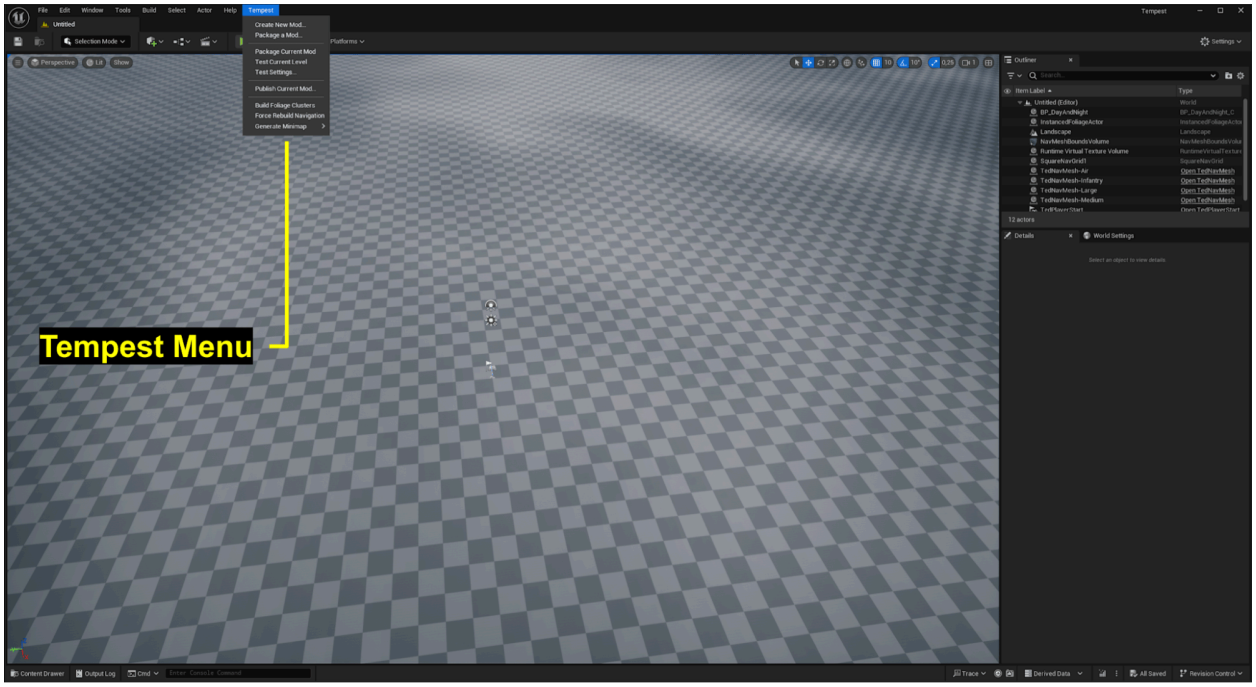
Don't worry - everytime we use a technical term, we'll make sure to show you exactly what / where we click. This is more to give you a small vocabulary for some key Unreal elements.

If you want to dive deeper, Epic Games has a 'Getting Started Guide' for Unreal Engine as well. Though this is not required - it is a good starting point if you are unfamiliar with Unreal Engine. And although Epic Games' guide will not show anything Tempest Rising specific – they do show how to get started with Unreal Engine, which is great know-how for playing around with the TR Map Editor tool.

Without further ado, let's get started!

# ‘Tempest Rising’ Top Menu

Inside the Map Editor project file provided to you, we’ve added a module that adds a “Tempest” button at the top of the editor window:




This menu allows you to:

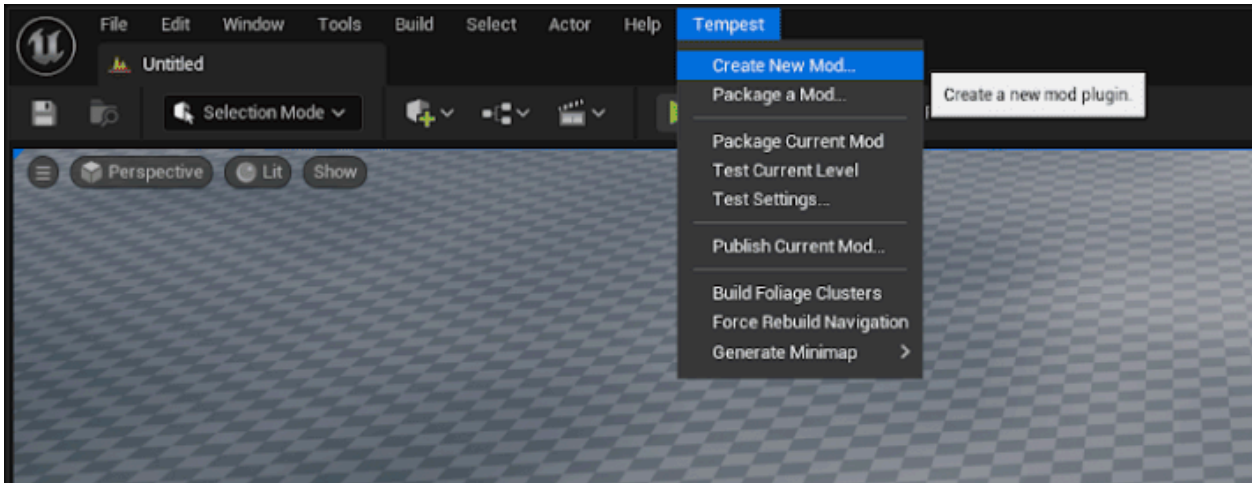
- **Create a New Mod**
  - *… creates an Unreal mod plugin with a Content Folder that houses your Mod / Maps, as well as any custom assets or blueprints you create or import*
- **Package a Mod**
  - *… ‘packaging’ means to export your Mod / Maps into a folder that you can add to Tempest Rising, or share with your friends. This button allows you to do it to any one of your Mods.*
- **Package Current Mod**
  - *… packages the Mod / Map you currently have open*
- **Test Current Level**
  - *… automatically packages and opens your map using the Tempest Rising .exe. The Map Editor project file does not contain any Tempest Rising code - so opening / owning Tempest Rising is necessary for testing.*
- **Adjust Test Settings (“Test Settings…”)**
  - *… ‘Test Settings’ allows you to set your default testing game mode, bot opponents, etc.*
- **Publish Current Mod (to Steam Workshop)**
  - *… relevant for sharing mods online; see main documentation for more info.*

- **Build Foliage Clusters**
    - *… relevant for Tempest Foliage; see main documentation for more info.*
- **Force Rebuild Navigation**
    - *… relevant for navigation tools; see main documentation for more info.*
- **Generate Mini-Map**
    - *… generates a mini-map image of the current map*

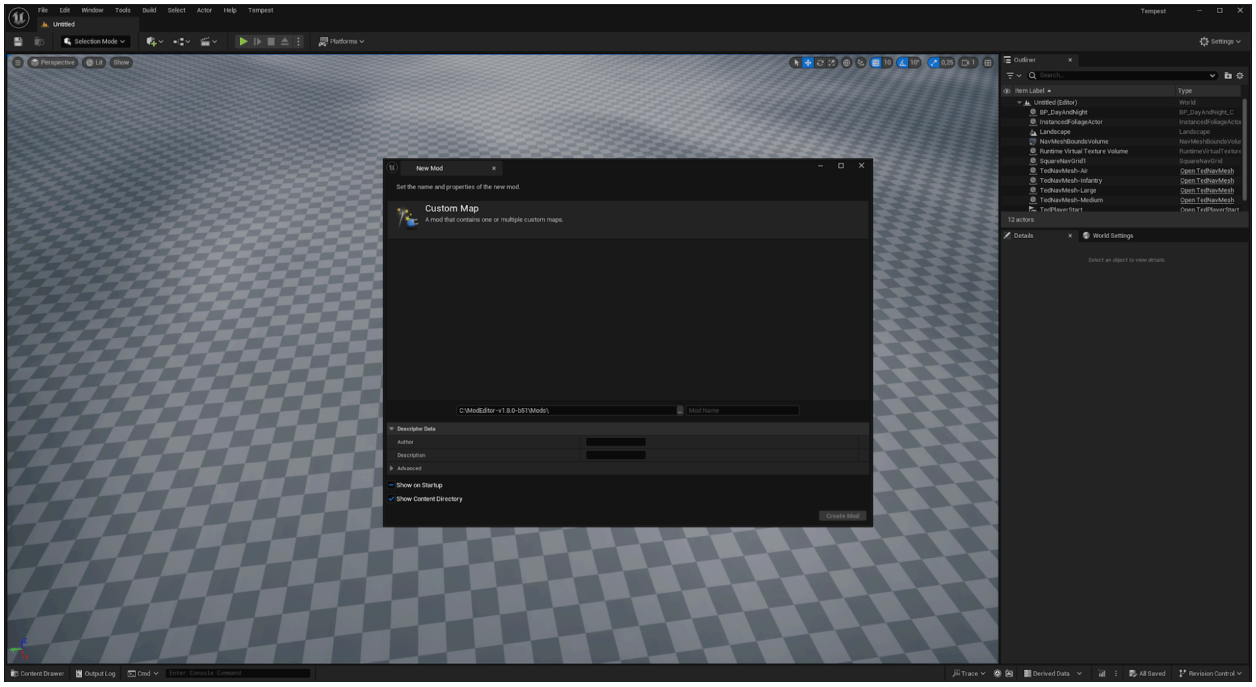
We'll go through the options that are relevant for creating your first map. The main documentation will describe each of these functionalities as well, so we'll just focus on the basics for now.

## Create Mod

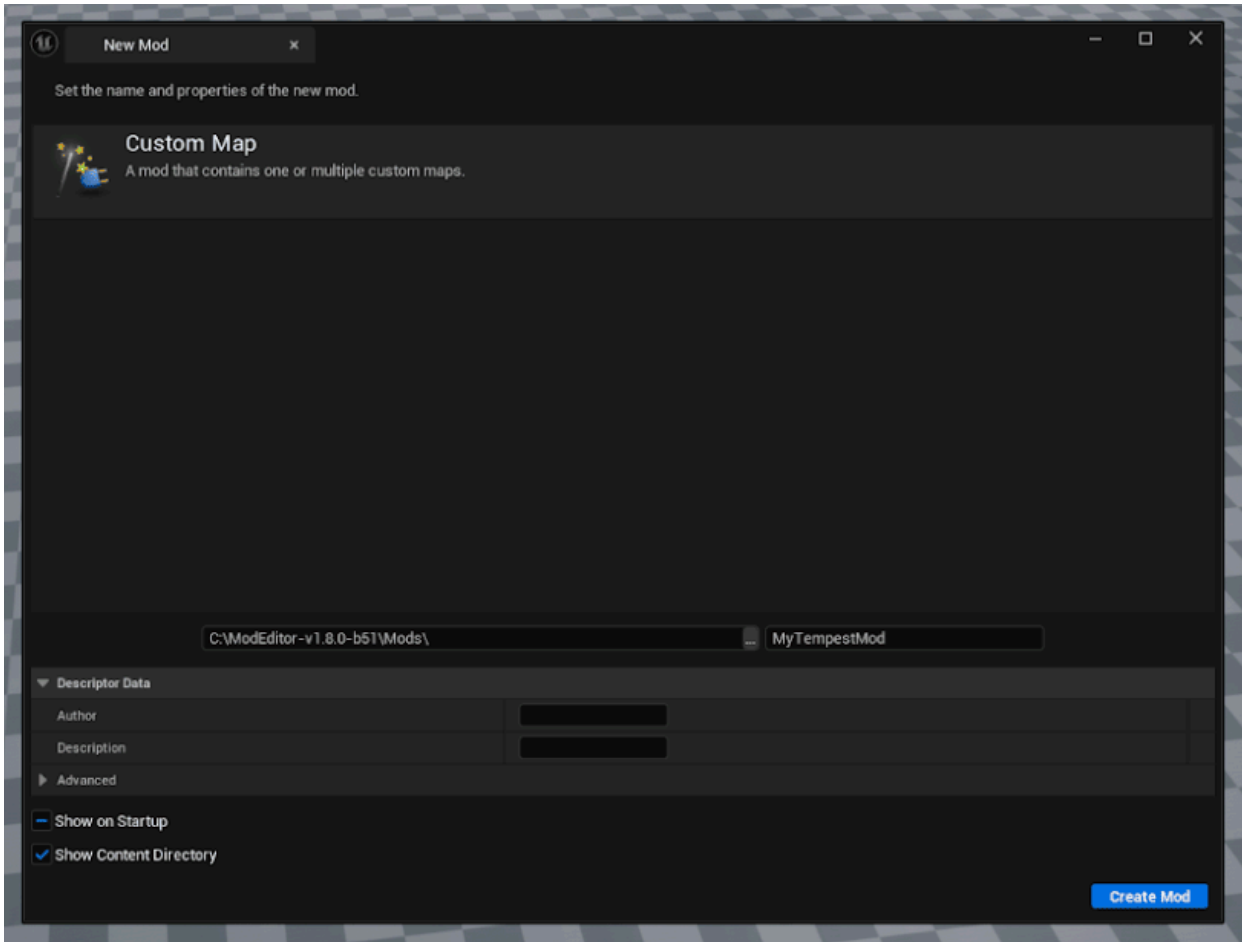
First up - **>CLICK<** 'Create New Mod' in the Tempest menu:




Now you should see the following window, where you can name your mod:



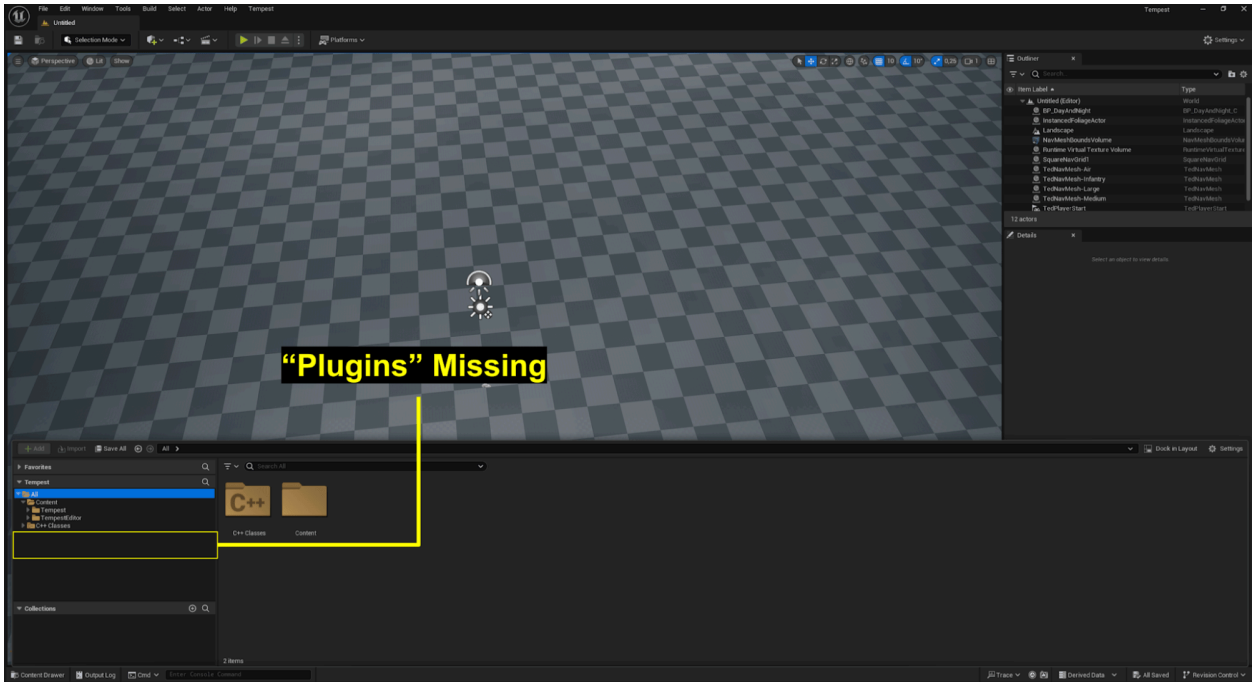

In this example, we'll name it "MyTempestMod". Once you've named your mod, **>CLICK<** 'Create Mod' in the bottom right corner of this window:




## Content Drawer

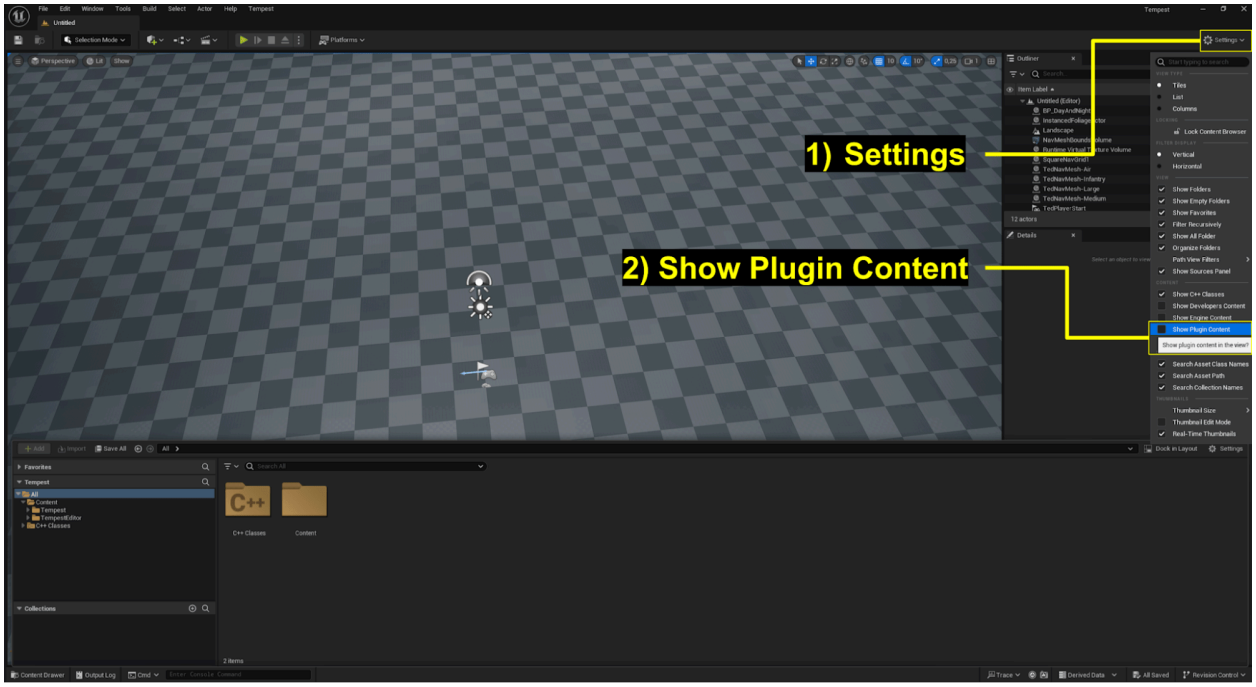
Your Mod has now been created, and a related **Content Folder** has been added to the project.

**BEFORE YOU CONTINUE: 'Plugins Content'** is not always set to visible by default with fresh installations of Unreal Engine 5. Example:




Here's how you enable visibility on 'Plugins Content'.

- First, find the '**Settings**' button to the top left of the Unreal view.
- **>CLICK<** on '**Settings**' to see the drop-down, and find 'Show Plugins Content'.
- Then, **>CLICK<** on '**Show Plugins Content**' to enable visibility on 'Plugins Content':

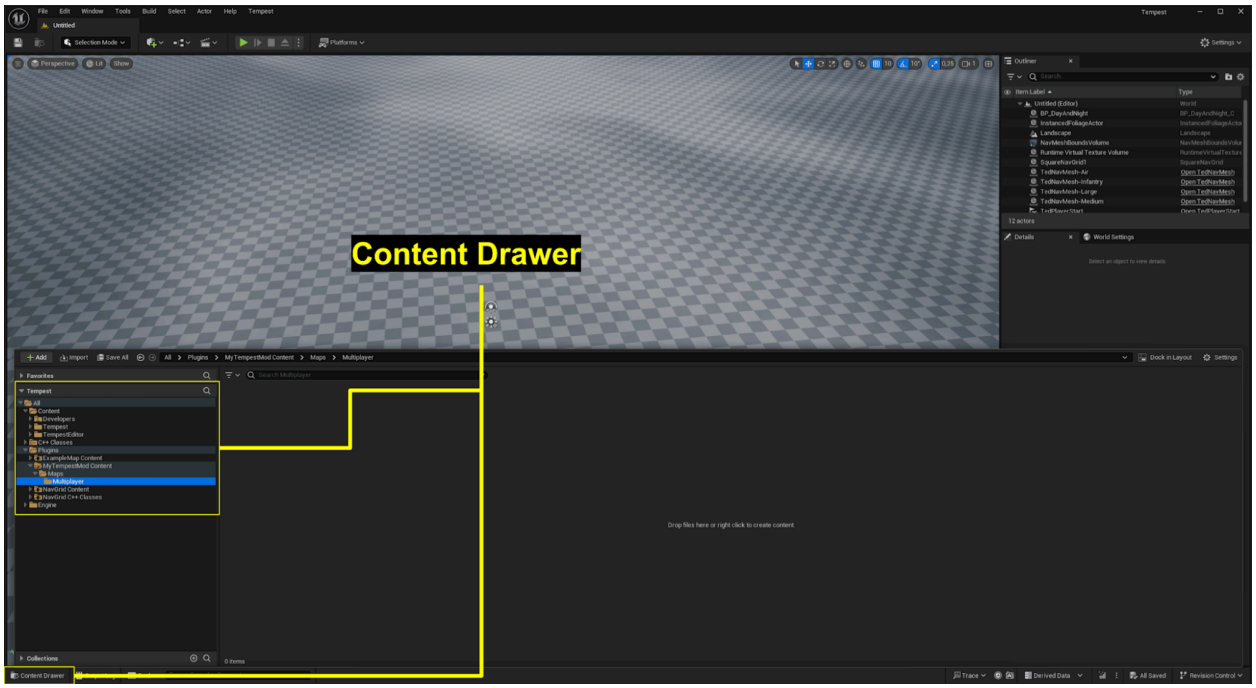


Now that you've enabled visibility on 'Plugins Content', you should be able to fold out the **Content Drawer** button, which you can find in the bottom left corner of Unreal.

**>CLICK<** on the Content Drawer button, and then **>CLICK<** until you fold out:

**Plugins → MyTempestMod Content → Maps → Multiplayer**

Once you've done this, you should be able to see the Content Folder inside of your mod:




ANYTHING that goes into this folder will get 'packaged' alongside your game. That means any assets you drag in here will be usable in maps that your mod contains.

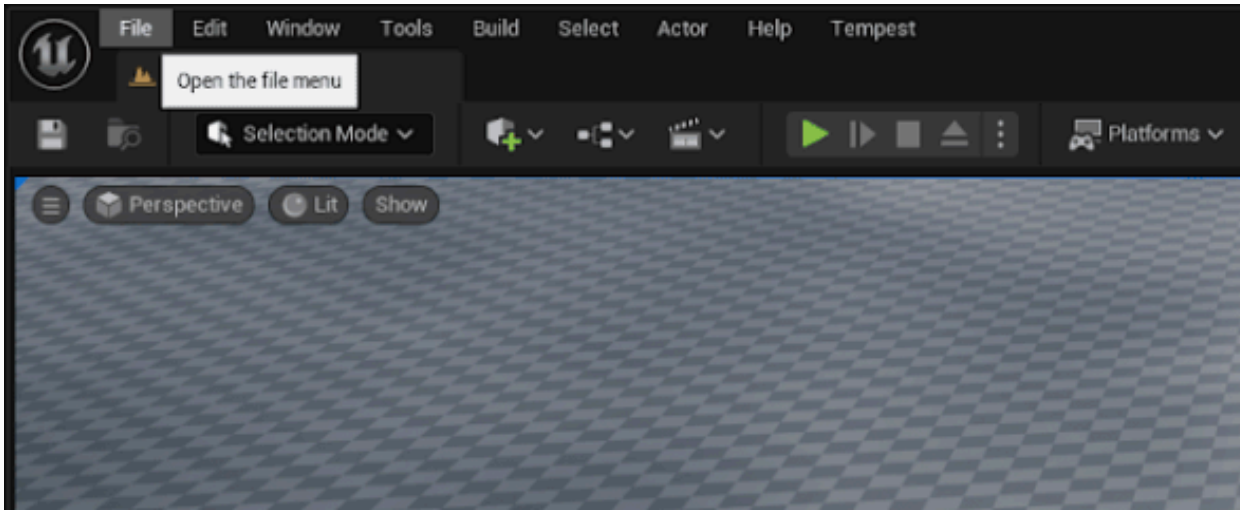
**NOTE**

*Note that a mod can contain multiple maps!*

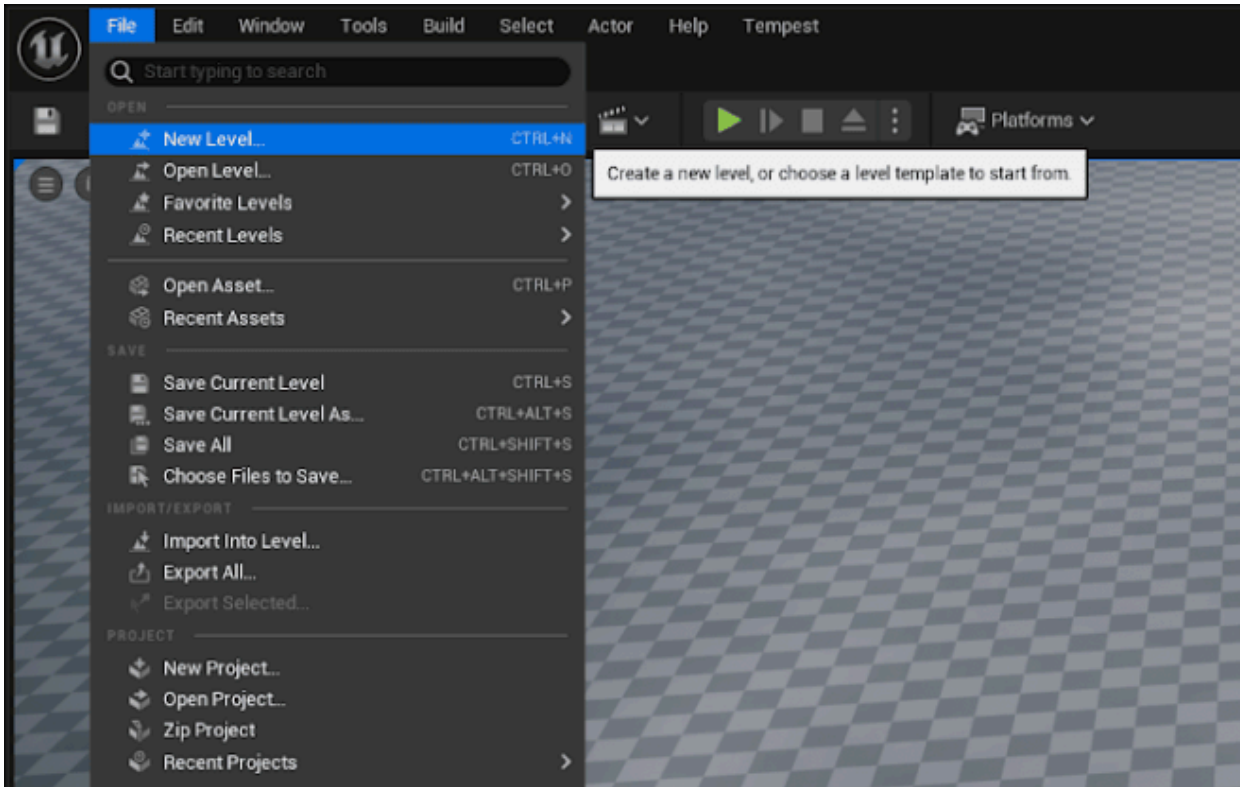
## Create New Level

Next up, we'll create a 'New Level'. This will be your first Tempest Rising map.

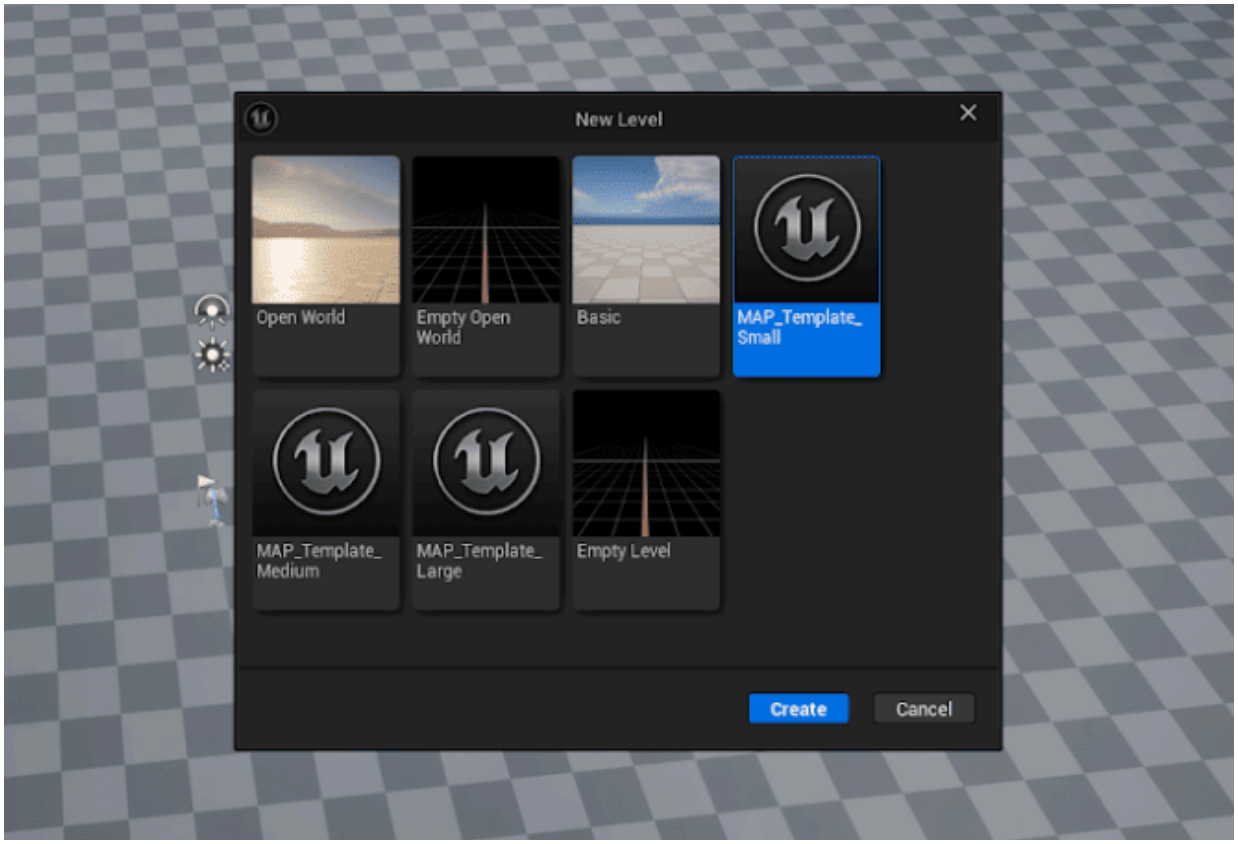
First, **>CLICK<** on 'File' in the Top Navigation Bar:




Then, **>CLICK<** on ‘New Level’:




Next, **>CLICK<** on ‘MAP_Template_Small’, and then **>CLICK<** on ‘Create’ in the bottom right:

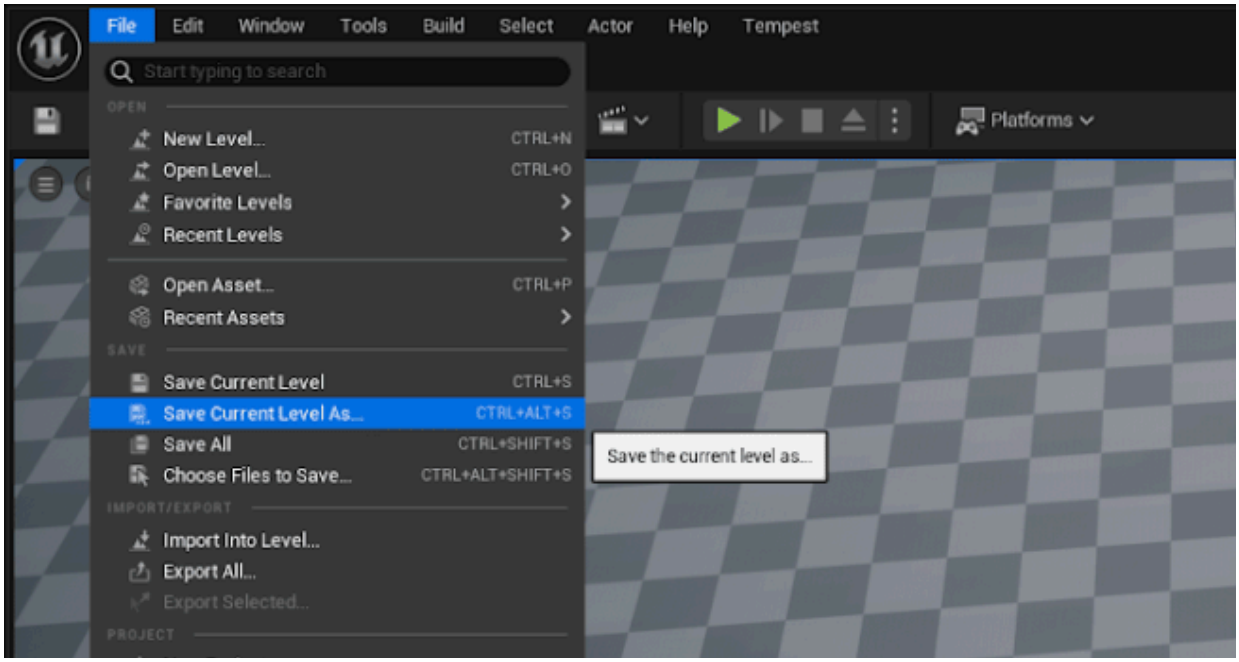



**NOTE**

> *Note that there are three template sizes: Small, Medium and Large. You can always change the size of your level later, by configuring the 'WorldBoundsActor', and extending your terrain mesh.*

You should now have a template level that you can save into your Content Folder.

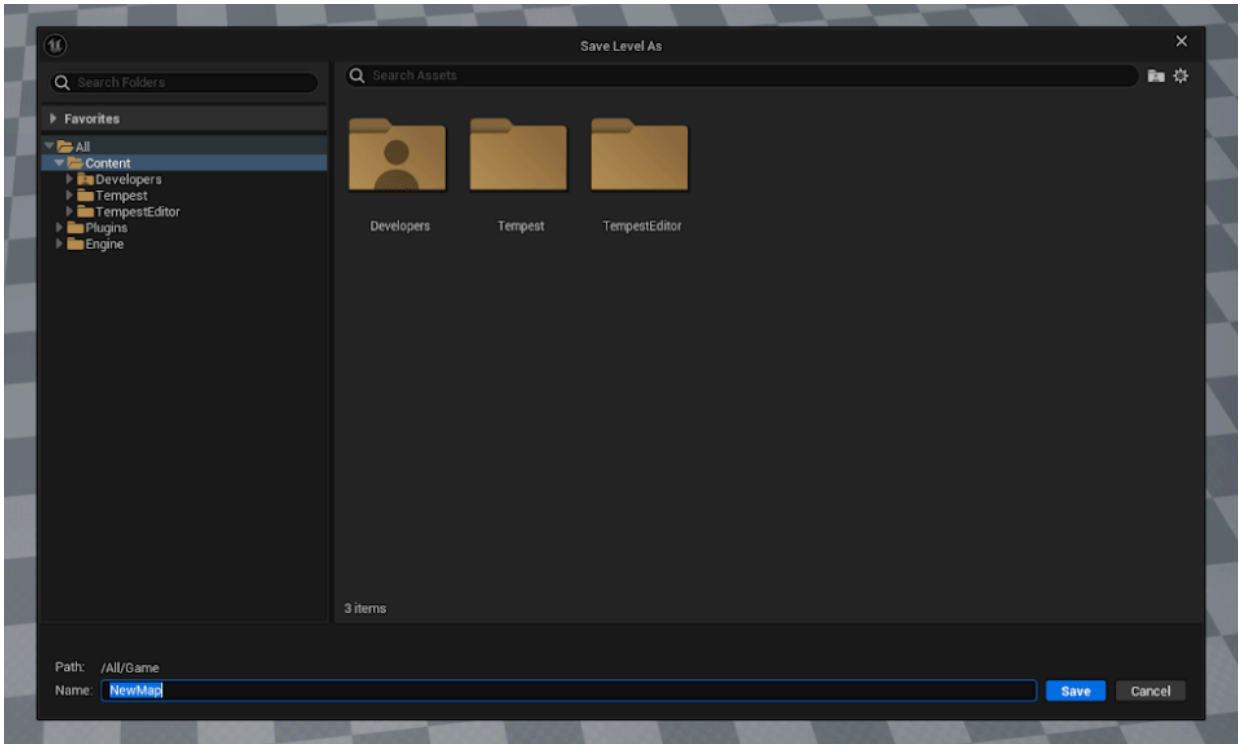
To do this, **>CLICK<** on 'File' again in the Top Navigation Bar, and then **>CLICK<** on 'Save Current Level As…'.




You'll be faced with the following window, at the 'top' of the Content Folder hierarchy. You'll need to find your Mod Content Folder in this window, in order to save your map there.

**NOTE**

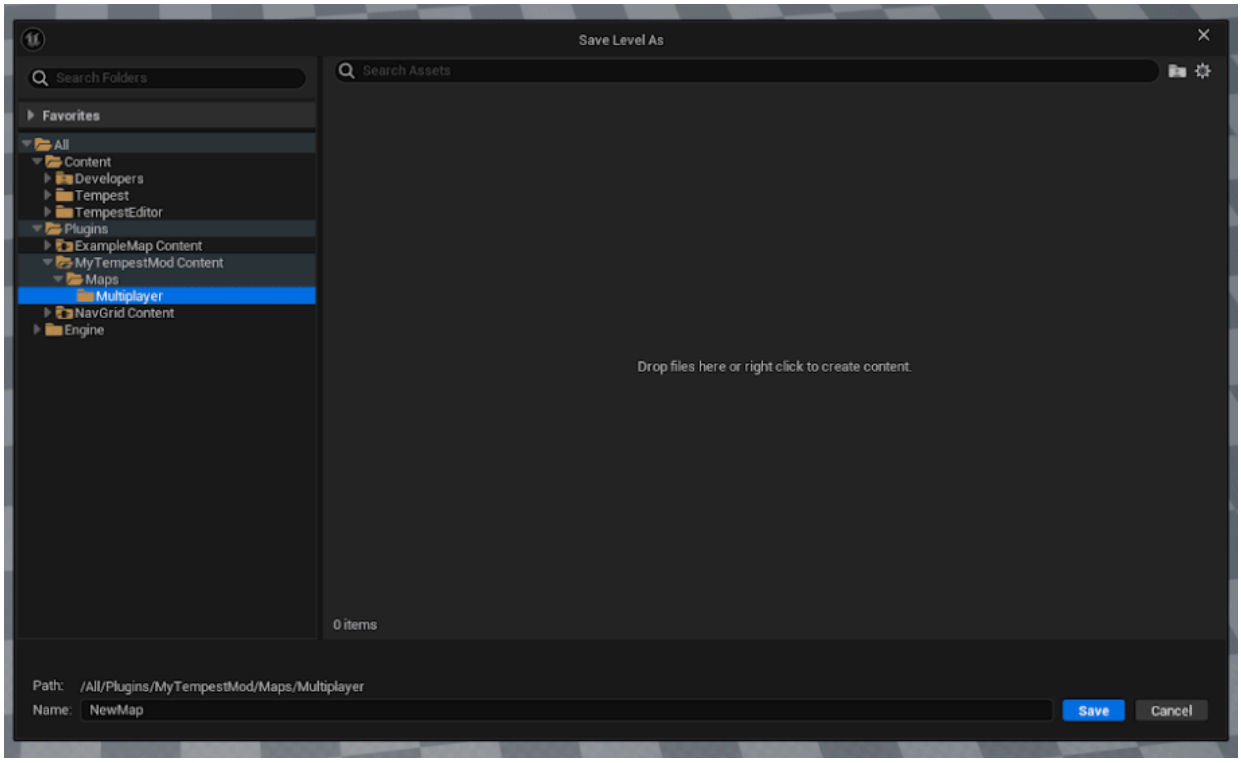
*You may not have the 'Developers' folder here. That's fine; you don't need it for this guide. The important folder is the 'Plugins' folder.*




To find your Mod Content Folder, follow the same steps as previously. **>CLICK<** and fold out:

**Plugins → MyTempestMod Content → Maps → Multiplayer**

Once you've done this, you should be able to see the Content Folder inside of your mod:




Now you should be able to name your level, and save it inside your Mod Content Folder.

**NOTE**

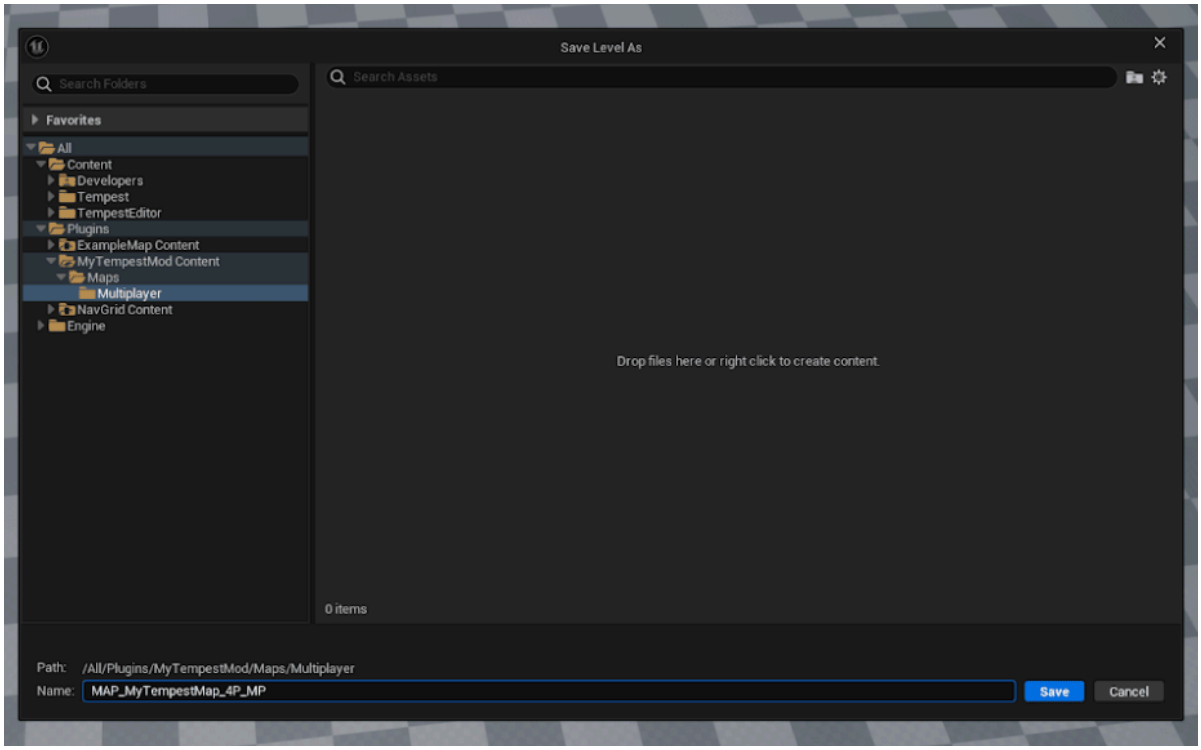
*Maps should be named MAP_MP_<MapName>_#P, where <MapName> is your chosen map name and # is the amount of players the map is for, for example MAP_MP_ExampleMap_2P.*

In this example case I've accidentally named my map incorrectly, but such is life sometimes. The naming doesn't matter too much; it just can't have an identical name to an existing Tempest Rising map.

To avoid this, preferably use a unique name.

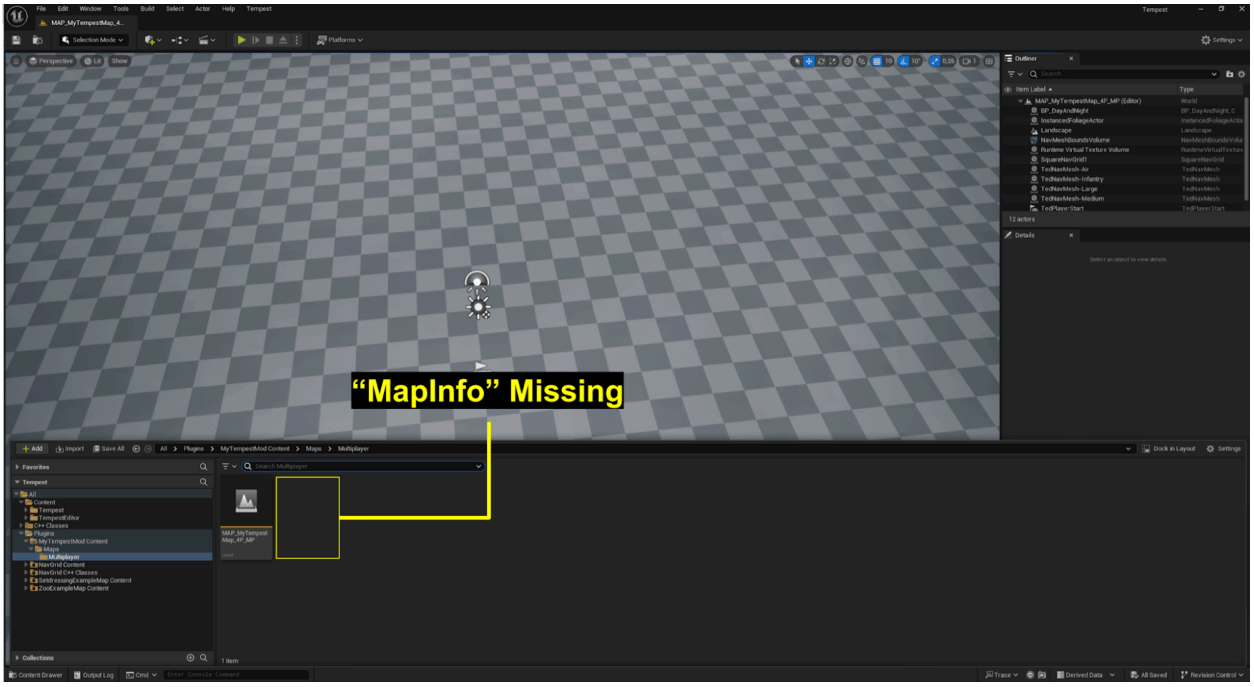
Also, try to avoid names that are extremely long, as Unreal Engine does not like long file names/file paths.

For the rest of this guide, the example map is called "MAP_MyTempestMap_4P_MP":



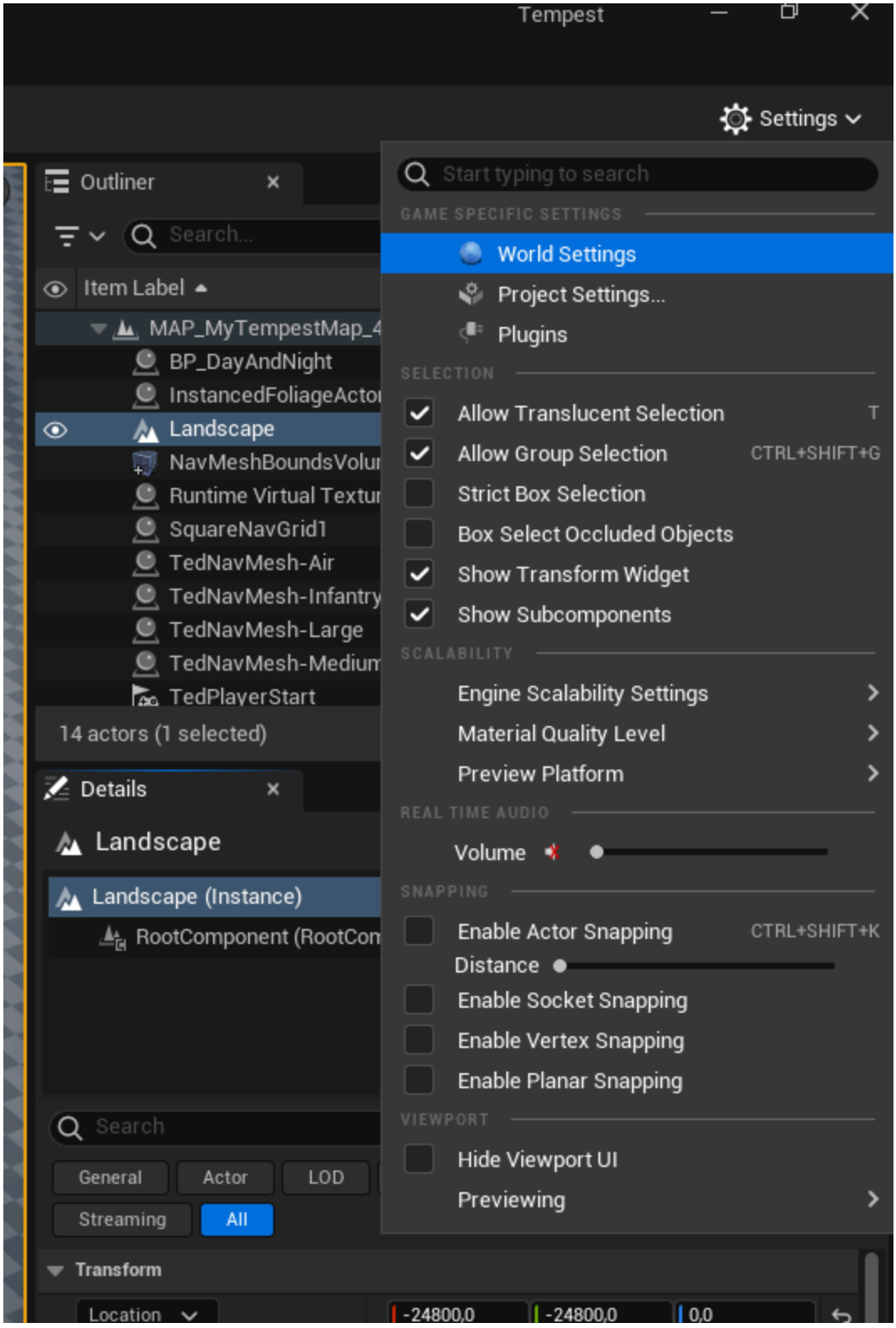

**>CLICK<** on ‘Save’ on the bottom right, and you will have saved your level into your Mod Content Folder.

Now, you’ll have to enable ‘**MapInfo**’ for your level. MapInfo is a type of data asset, which allows you to configure different variables, like player spawns, player count, map name, etc.. If you look in your Mod Content Folder, you’ll see that the MapInfo asset is currently missing:

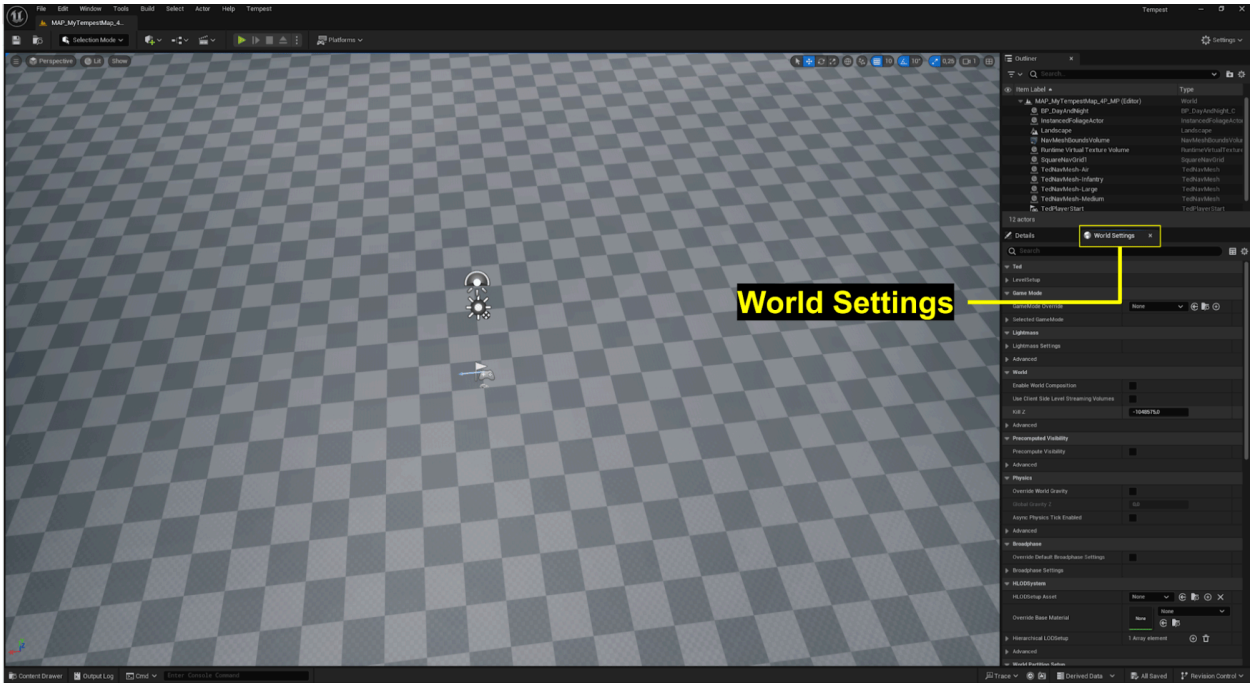



To enable '**MapInfo**', first you need to enable the **'WorldSettings'**, if it's not already enabled.

To enable '**WorldSettings**', **>CLICK<** on the '**Settings**' on the top right of the Unreal view:



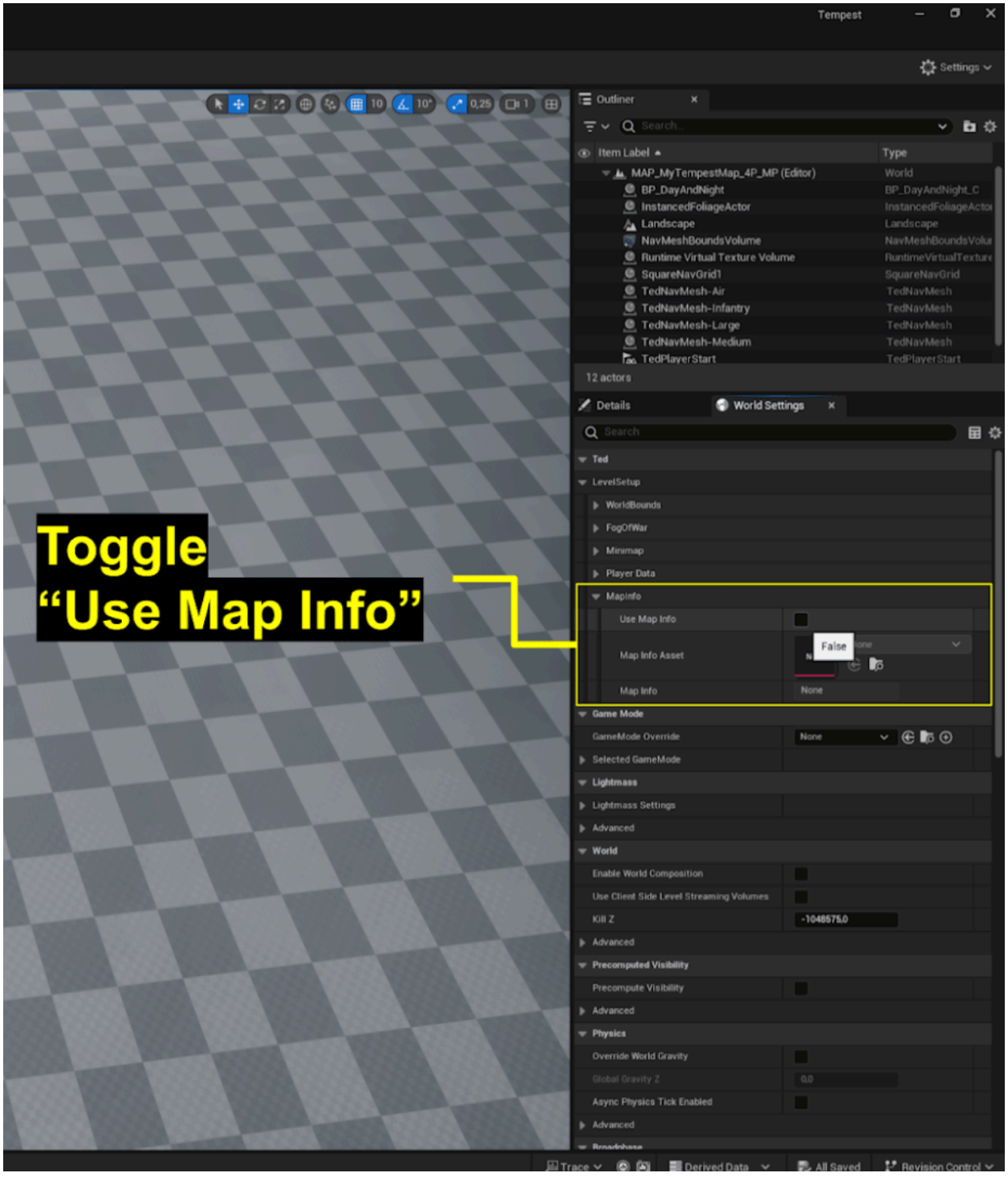

When you then **>CLICK<** on the World Settings panel to the right in the Unreal view, this is the first view you will see:



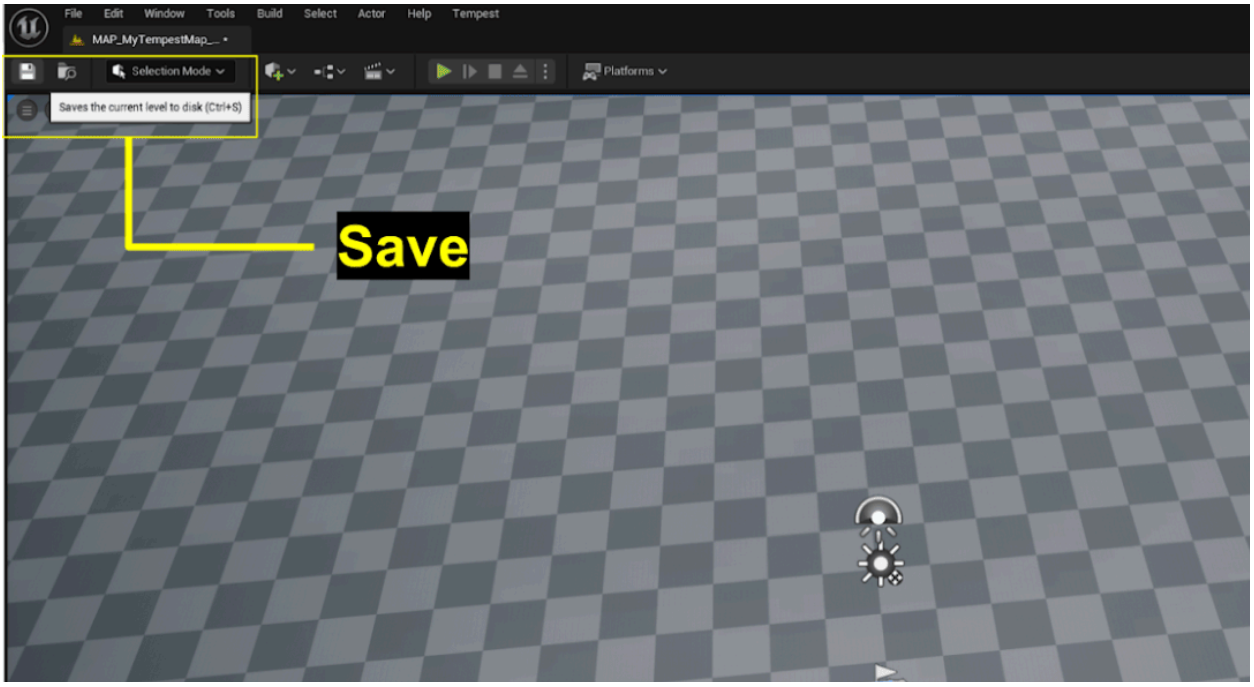

From here, **>CLICK<** on and fold out:

**LevelStartup → MapInfo**

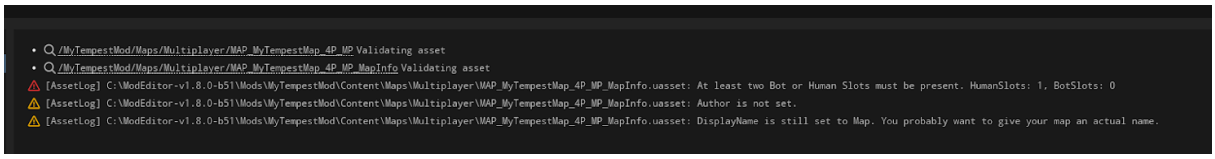
You'll then see the following view, where you need to **>CLICK<** on '**Use Map Info**', and toggle it on:




Once you've done this, **>CLICK<** on '**Save**' in the top right corner of the Unreal view:

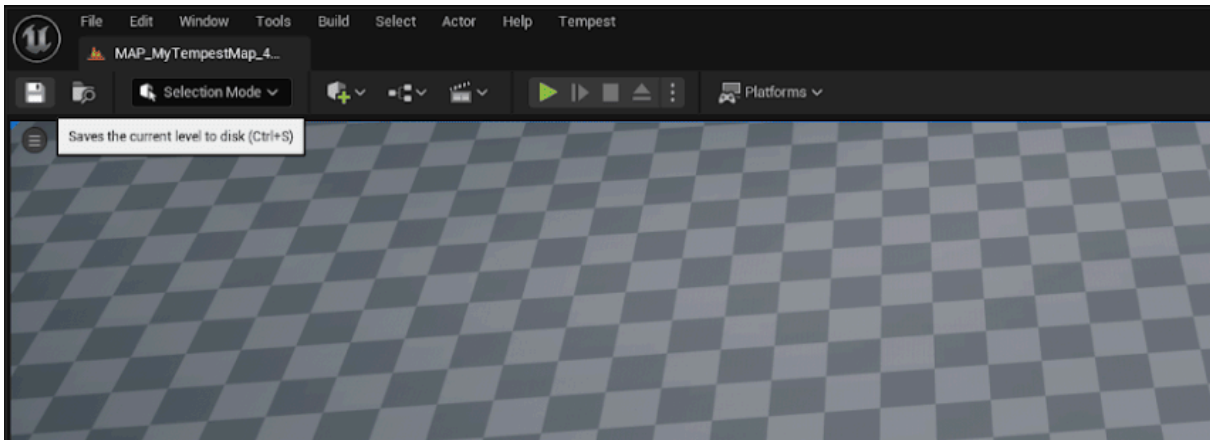



After saving your level for the first time, you may see this error message:

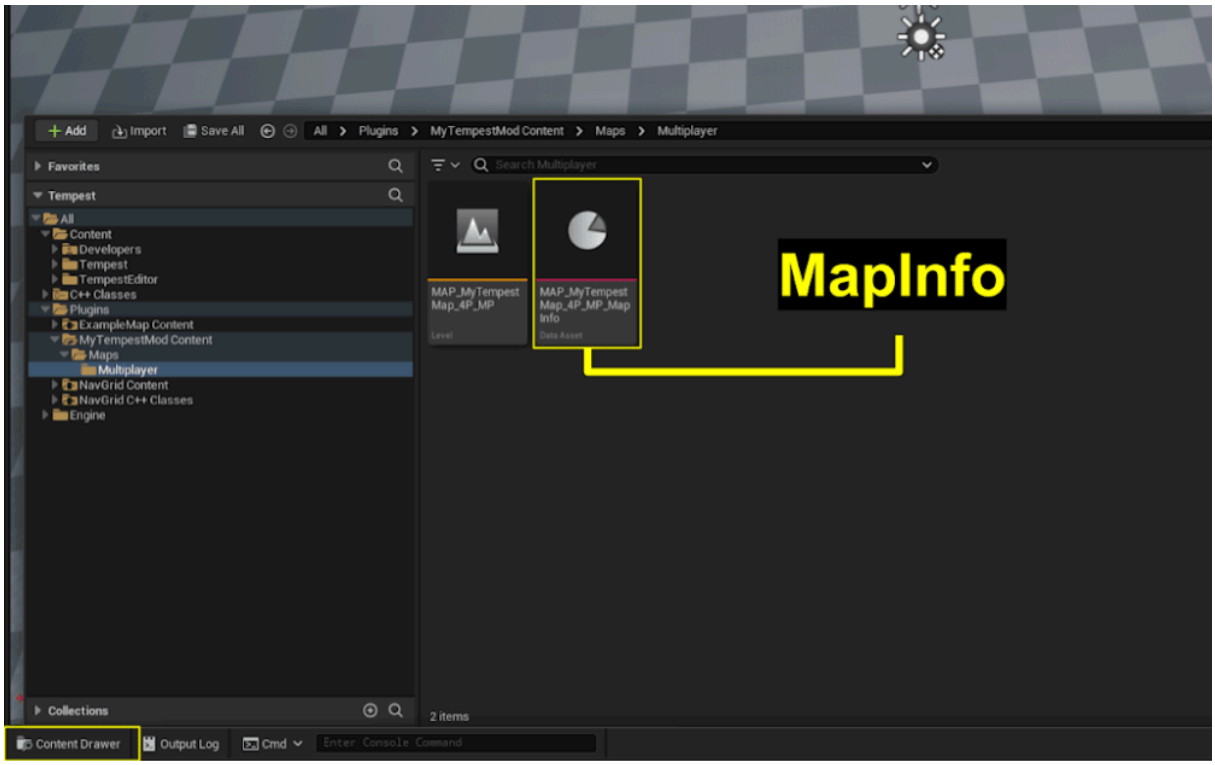



This is to be expected. Since you are saving a map for the first time, no **MapInfo** exists, and **TedPlayerStarts** have not been configured yet. These errors are safeguards to remind Modders to configure and include these two crucial elements.

To fix the red error, **>CLICK<** ‘Save’ in the top left corner again (or press CTRL+S):



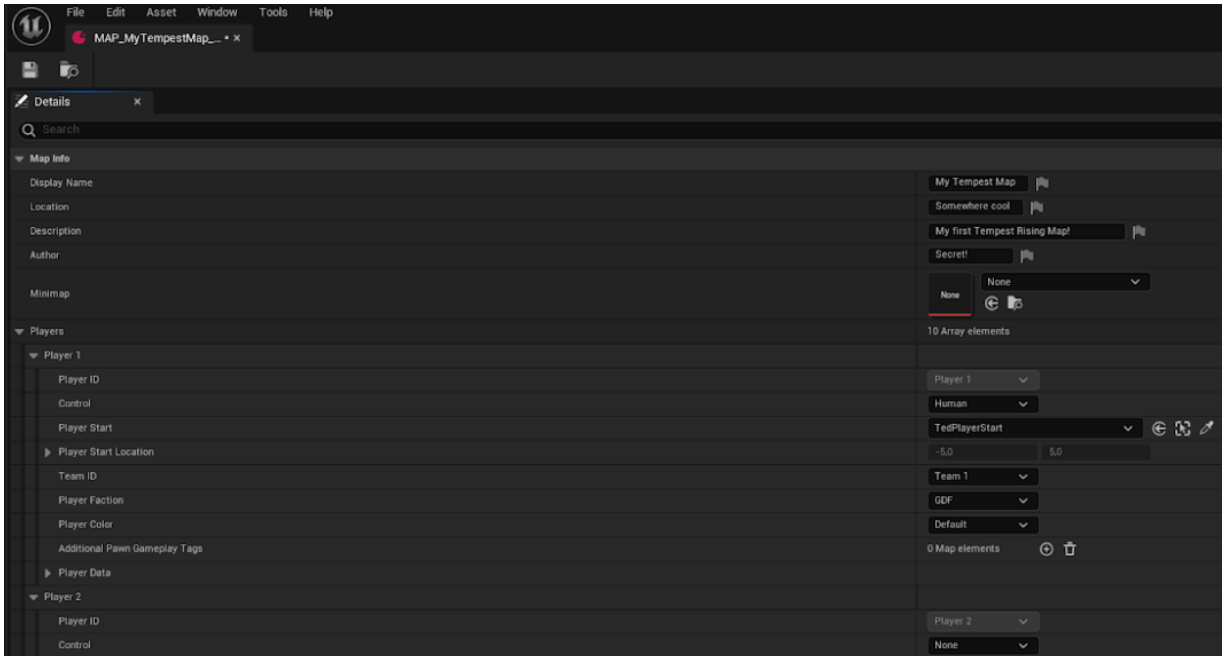
This will automatically generate a **MapInfo** asset for your level, and add it to your Mod Content Folder next to your level file:




## Configure MapInfo

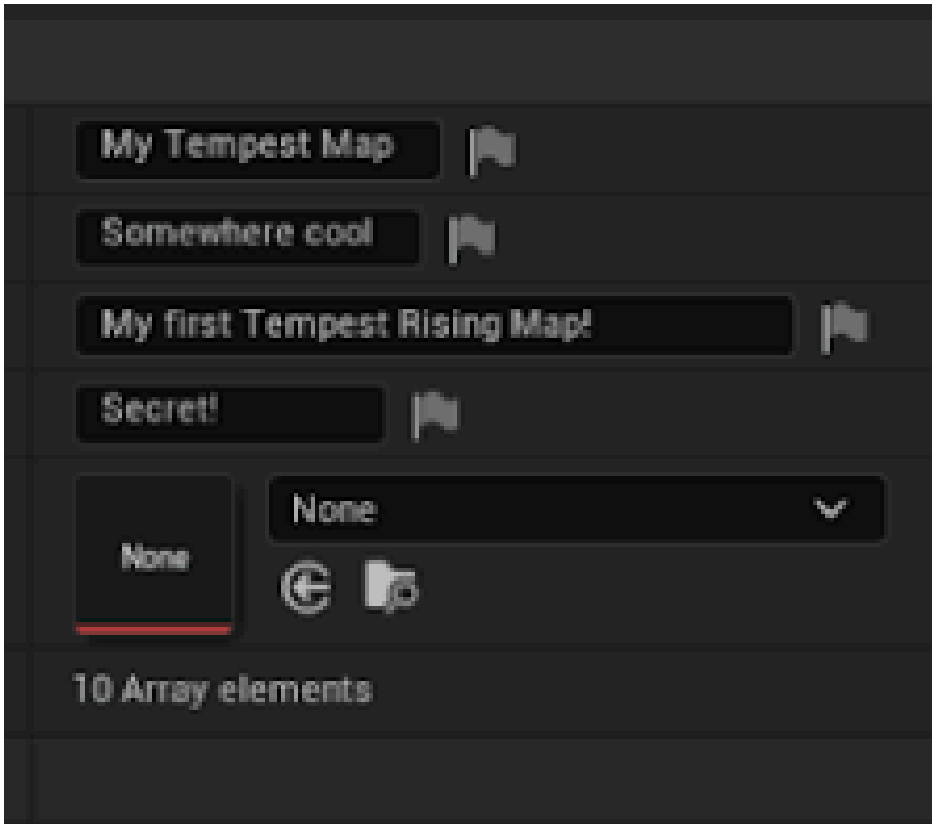
The **MapInfo** asset is a collection of information on your level. This is where you define:

- Map Name (To be displayed in-game, in Skirmish/Custom Games)
- Location (To be displayed in-game, in Skirmish/Custom Games)
- Description (To be displayed in-game, in Skirmish/Custom Games)
- Author (To be displayed in-game, in Skirmish/Custom Games)
- Player Starts (How many players, what TedPlayerStarts are used, etc.)
- Starting Armies (What starting armies are available / default)
- What gamemodes are supported (SP, MP, Sandbox)
- Also:
    - Victory track from the Jukebox (overrides default Victory track)
    - Defeat track from the Jukebox (overrides default Defeat track)




To start, off, I've **>CLICKED<** inside the top fields, and added my:

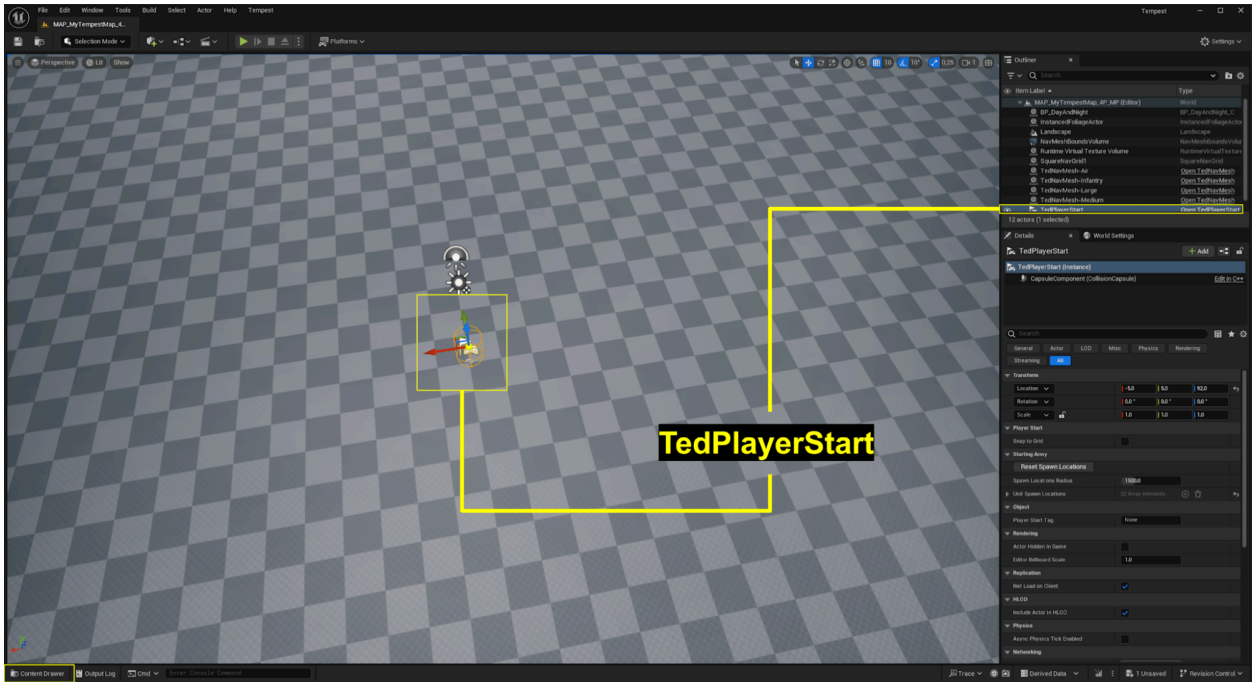
- Map Name
- Location
- Description
- Author




There's two things that we need to set-up, before we can fully fill out the MapInfo and package. These are **Generating a Mini-Map**, and **Setting up Ted Player Starts.**

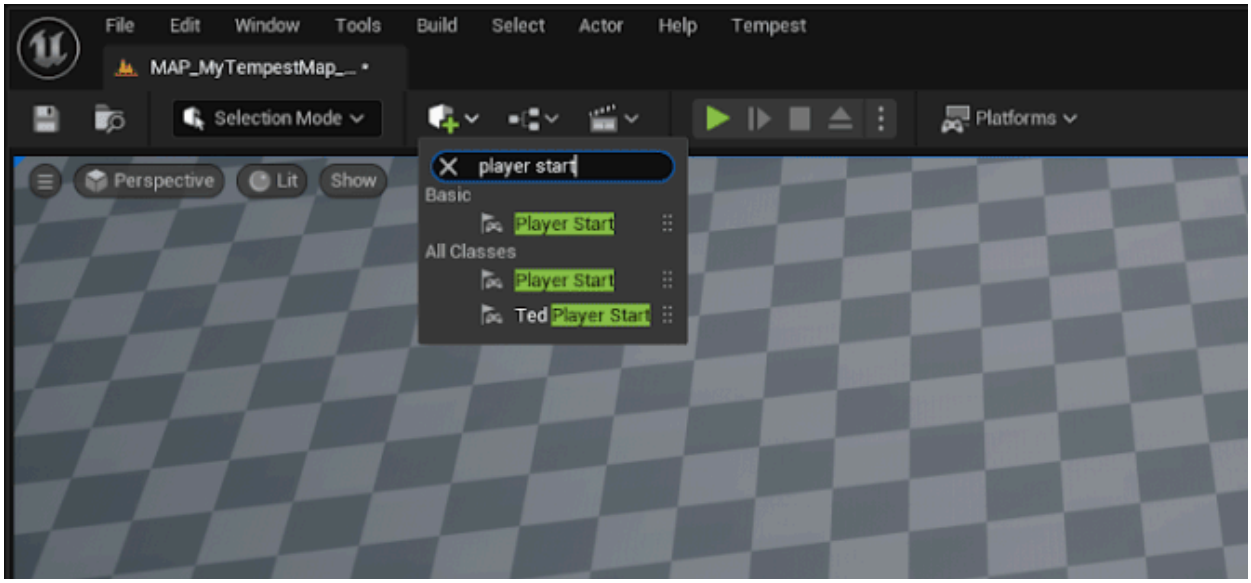
## Setting Up Ted Player Starts

If you have a look in your **Level Scene** in the background, as well as the **Outliner**, you'll notice that you already have 1 **TedPlayerStart** in the scene:



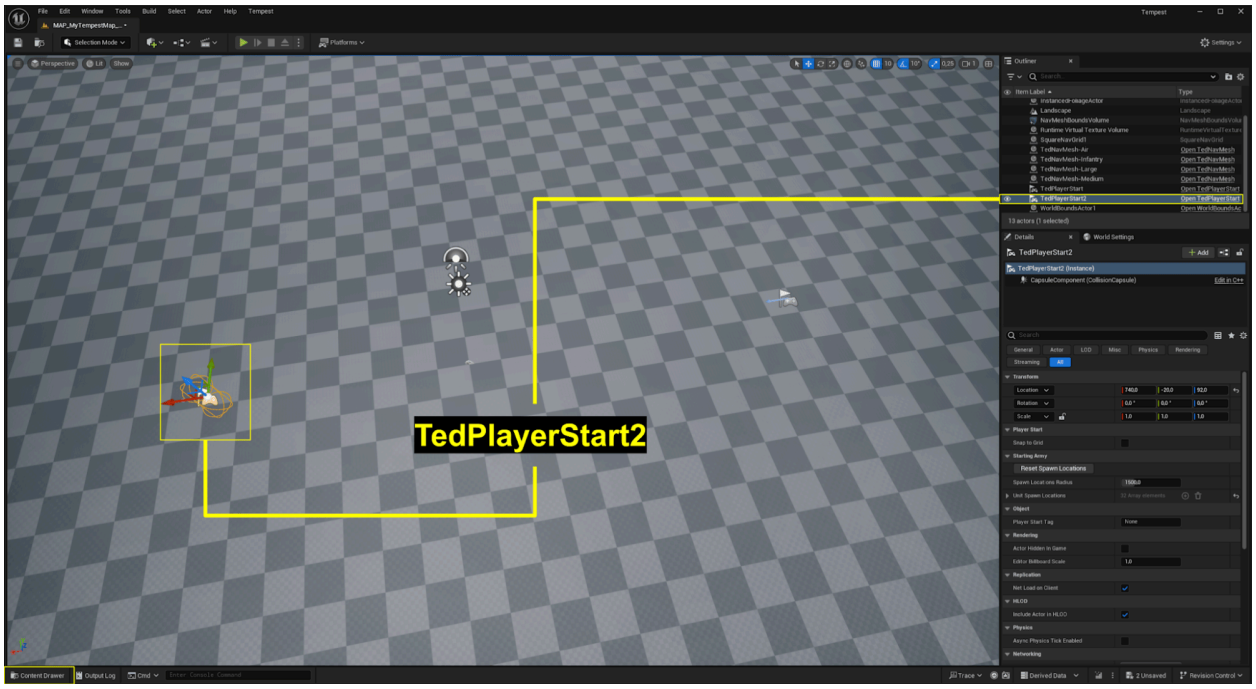

We need to add a second one to set up an opponent. We can either copy the existing TedPlayerStart actor or use the Place Actors Panel.

**>CLICK<** on the Place Actors drop-down in the top bar, and search “**Ted Player Start**”:



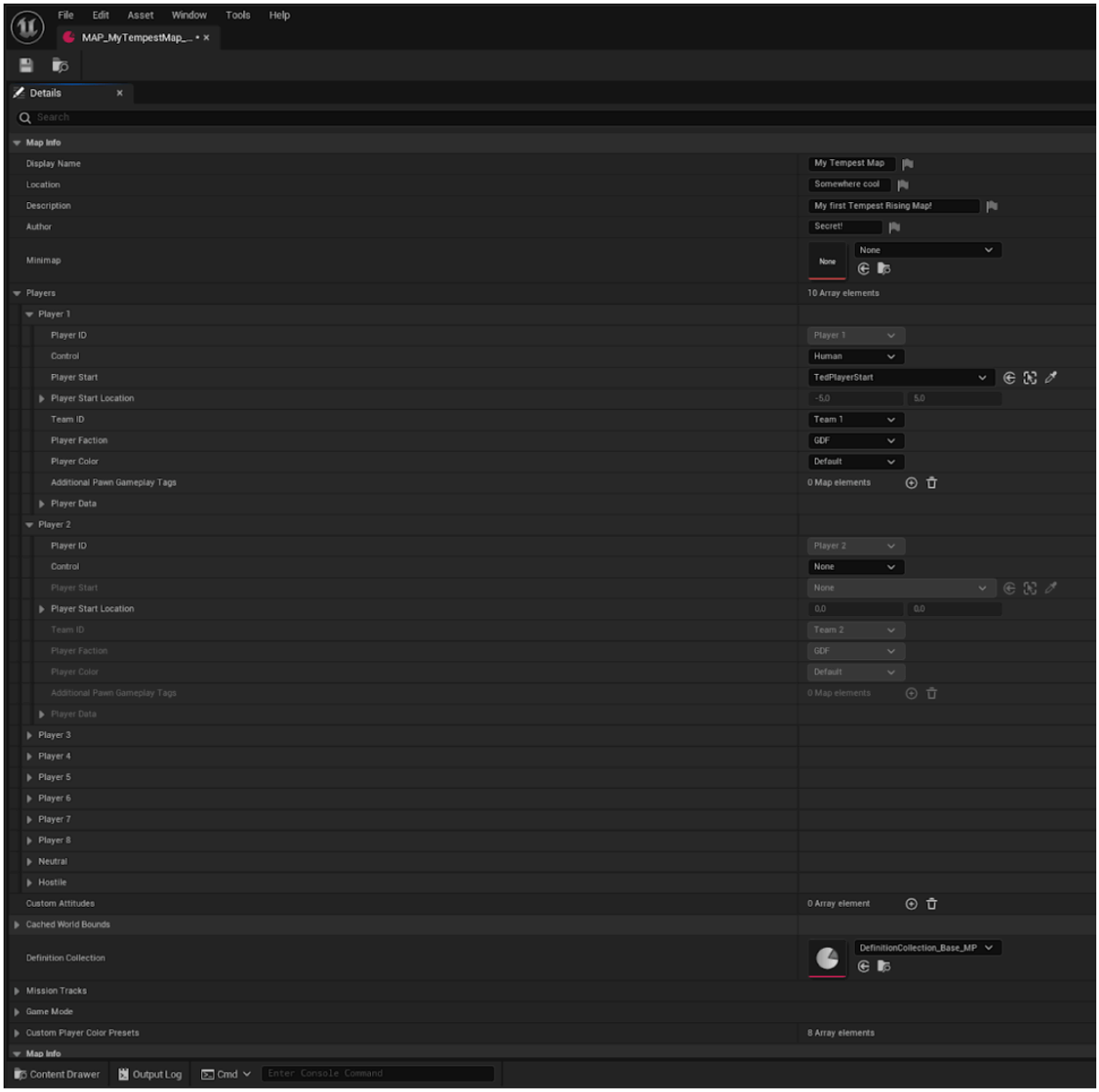

It’s important to use the **Ted Player Start** actor here, since the normal, default Unreal Engine Player Start will **NOT** work for Tempest Rising.

Next, >CLICK< on a **Ted Player Start** in the drop-down menu and >DRAG< it into the scene. This will create 'TedPlayerStart2', which should now be in your scene, and listed in the Outliner:



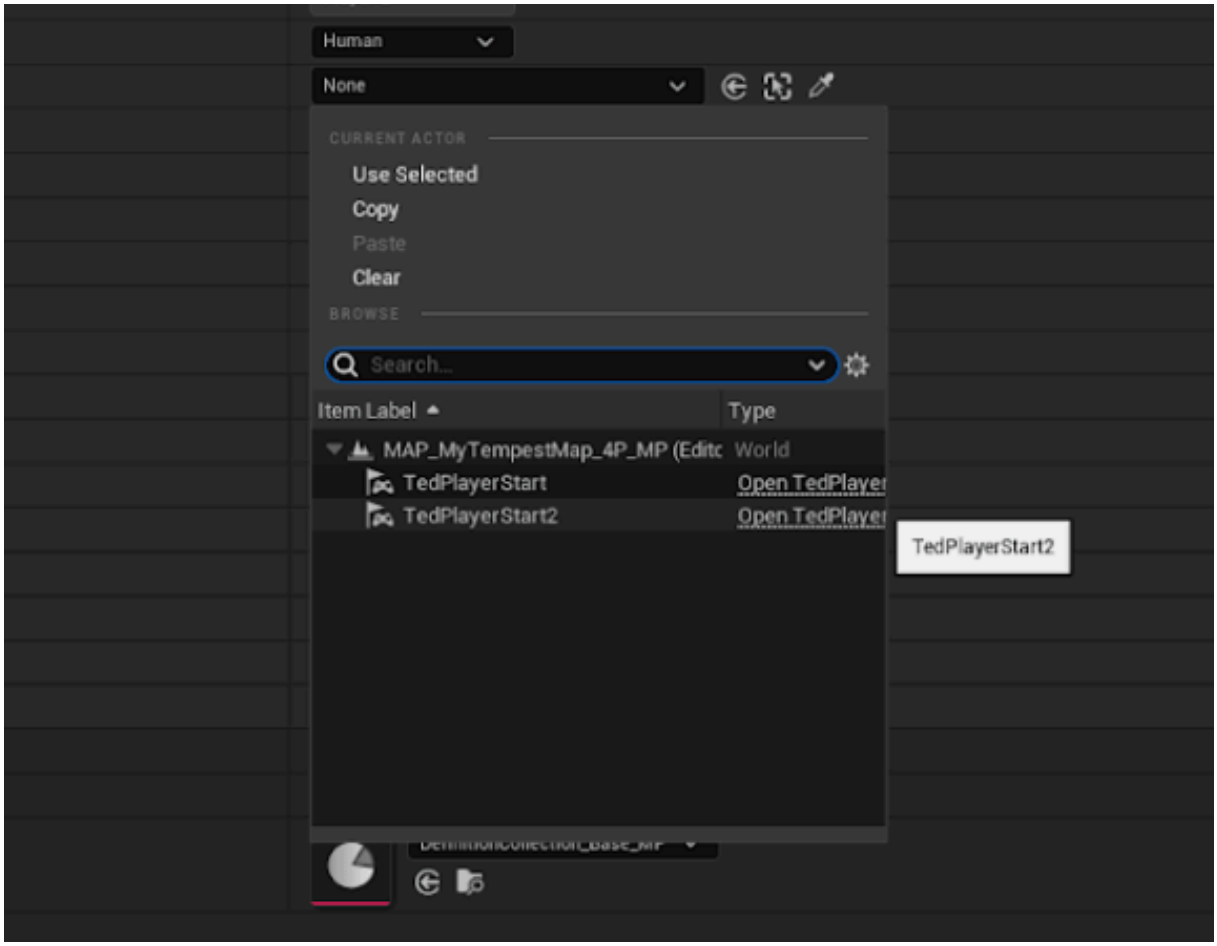

Now we can return to our **MapInfo**, and configure the player spawn for Player 2.

Double->CLICK< on your MapInfo to open it again, and >CLICK< to fold out the **Player** options as follows:

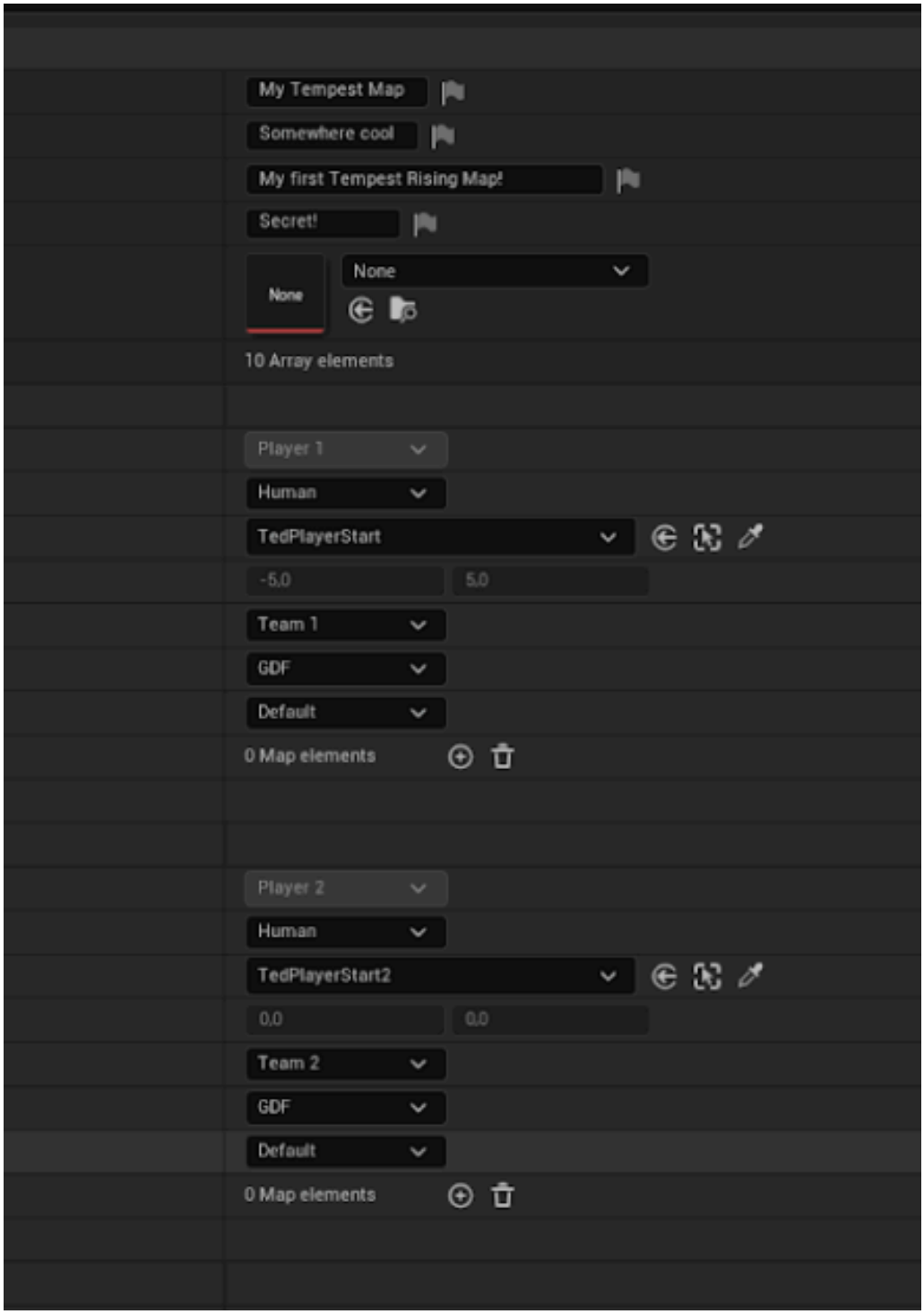



Next, **>CLICK<** on the **Control** drop-down, under Player 2. Here, you can select **Human**.

Then, **>CLICK<** on the **Player Start** drop-down. Here, you can now select **TedPlayerStart2**, since you've added it to your Level Scene:




Player 2 should now be set similar to Player 1, as seen below:

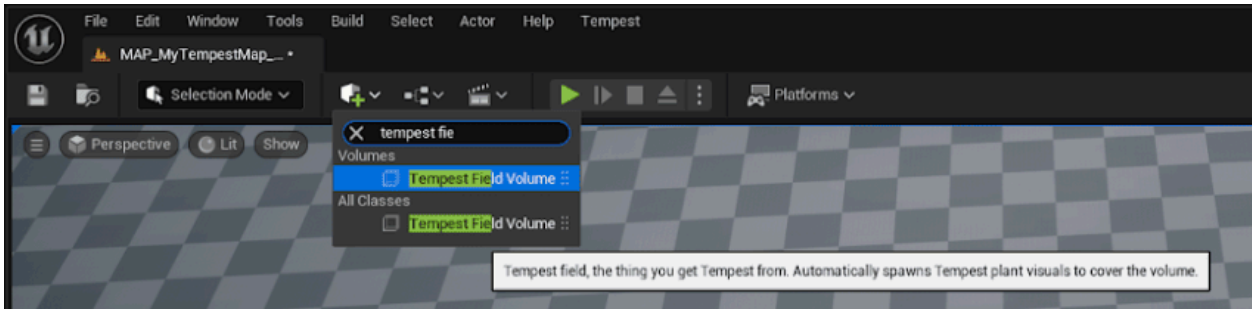



## Place a TempestField

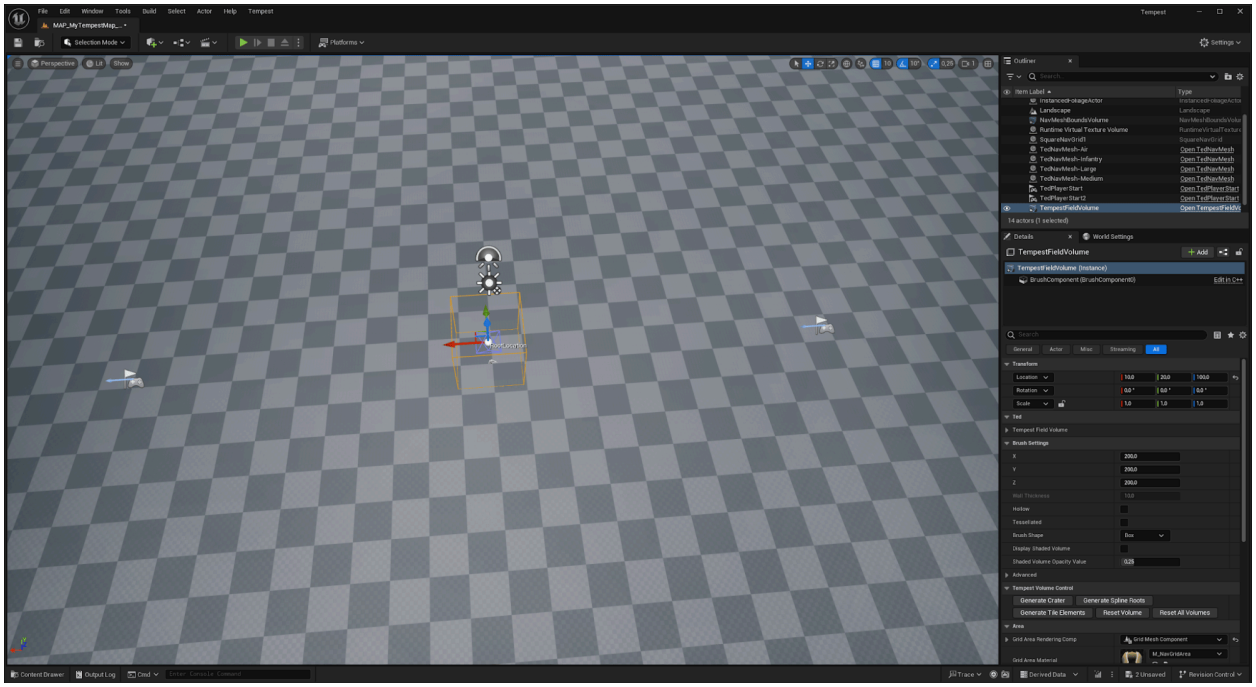
Before we generate a mini-map image, let's add a **Tempest Field** actor to the Level Scene. Tempest Fields are automatically visualized in the mini-map, when the mini-map screenshot is generated.

To do this, we follow the same steps as when adding a TedPlayerStart.

First, **>CLICK<** on the Place Actors Panel in the top left. Here, search for ‘Tempest Field Volume’:

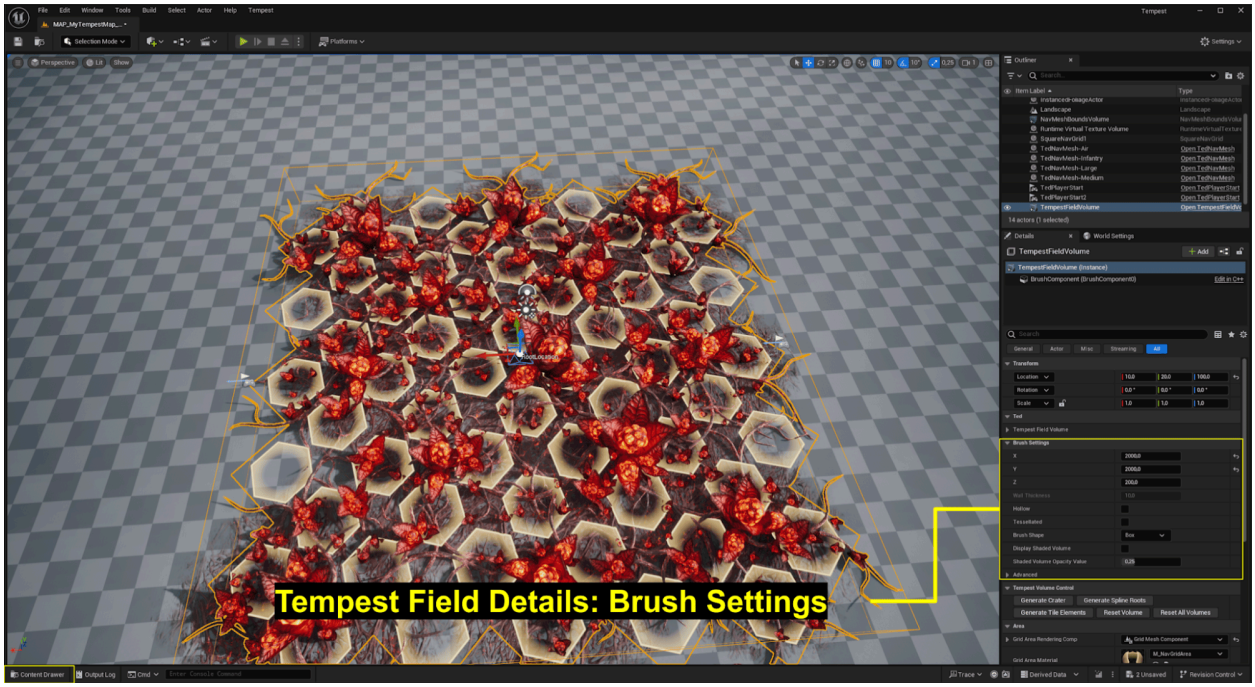



Then, **>CLICK<** on the Tempest Field Volume and **>DRAG<** it into the Level Scene. You should now have a Tempest Field Volume in your level (represented by a box):



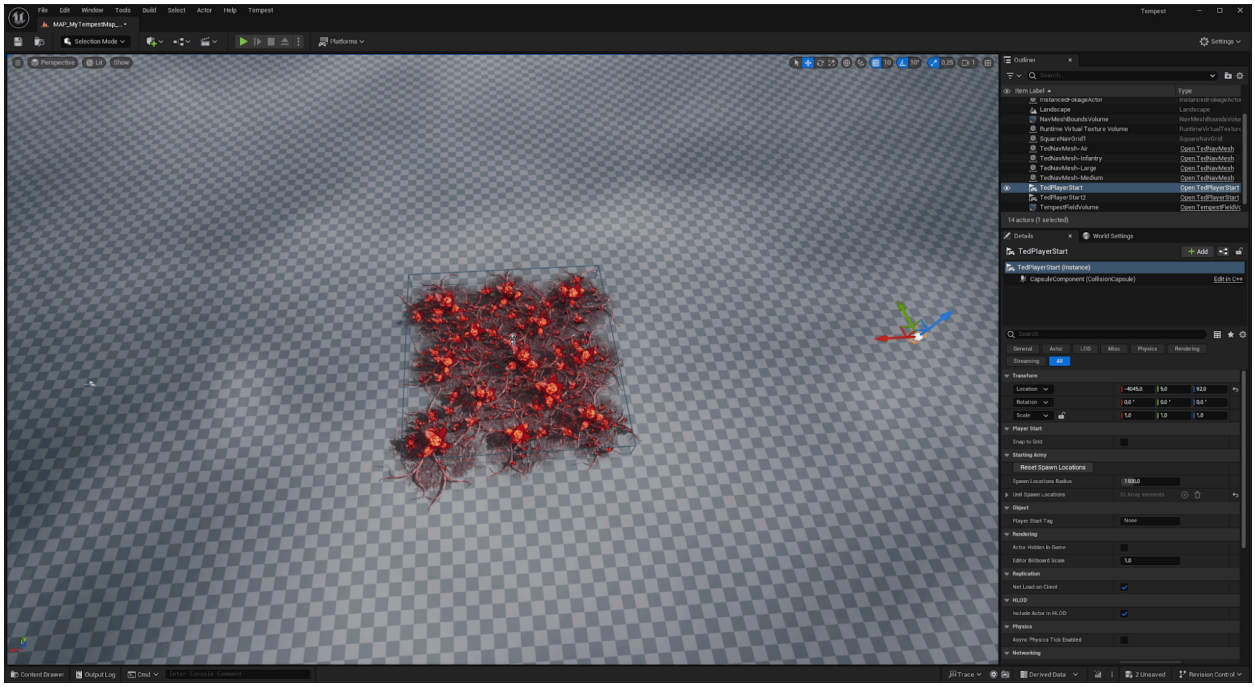
Next, **>CLICK<** on the Tempest Field Volume inside your Level Scene. You can now set the size of the Tempest Field (among other things, like adding a Crater that makes it automatically regenerate).

**>CLICK<** on the Details Panel to the right (if it’s not currently open), and find the **Brush Settings**. **>SET<** both the X and Y values to 2000. You’ll know it worked if your Tempest Field spreads out like this:




Now, I placed my TedPlayerSpawns a little too close to the Tempest Field.

So I've zoomed out a bit, **>CLICKED<** on each of my TedPlayerSpawns, and **>DRAGGED<** them a bit further apart (by dragging the little arrows that show up when they're selected):

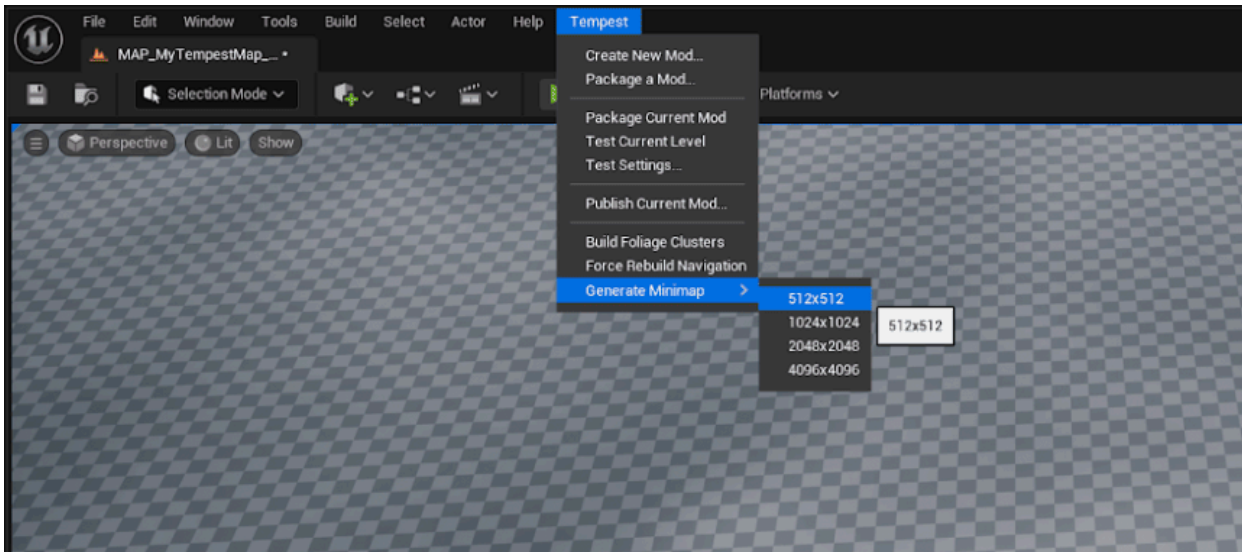


I've now successfully added a Tempest Field in between my two player spawn locations.

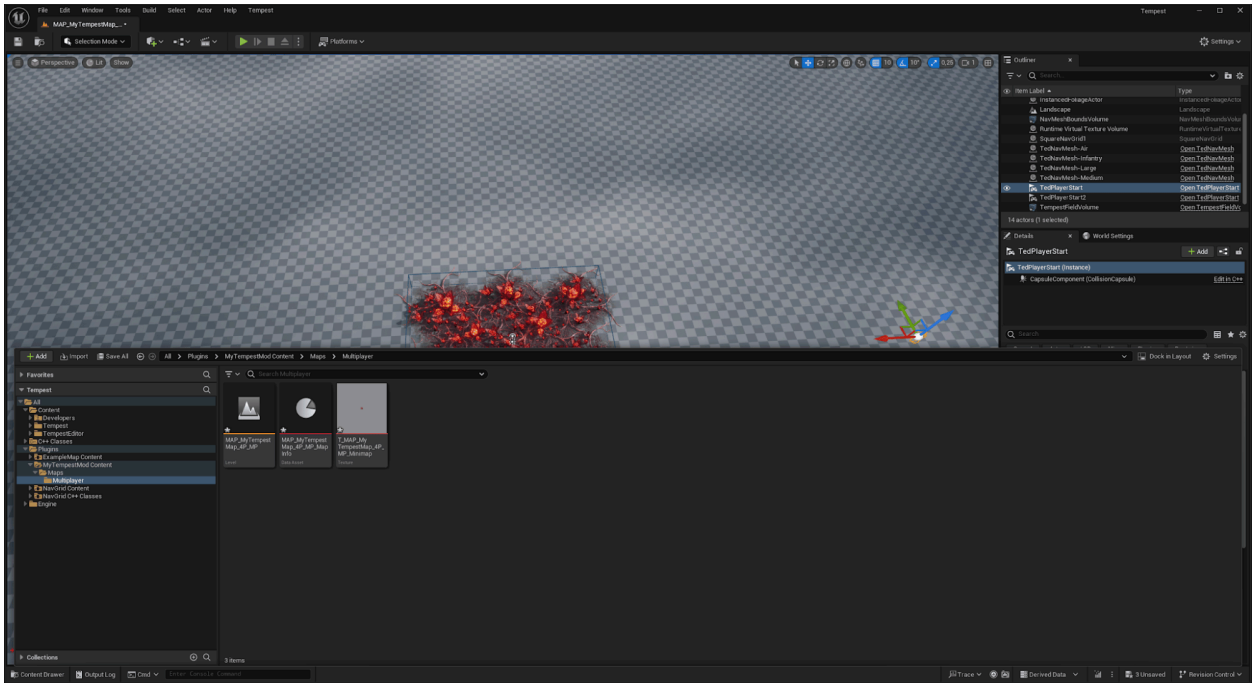
# Generate a Mini-Map Image

Generating a mini-map image is super quick.

Simply **>CLICK<** on the ‘Tempest’ menu in the Top Navigation Bar, and **>CLICK<** on ‘Generate Mini-Map’. **>CLICK<** on 512x512 to generate the mini-map:



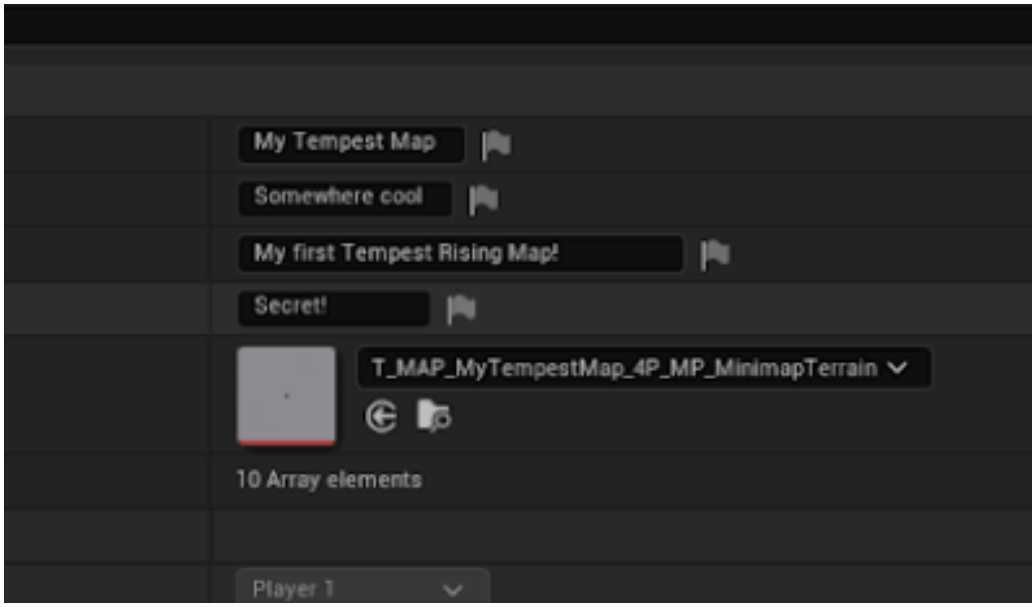

If you take a look in your Mod Content Folder, you’ll see that the Mini-Map has been added here as a 2D texture file:



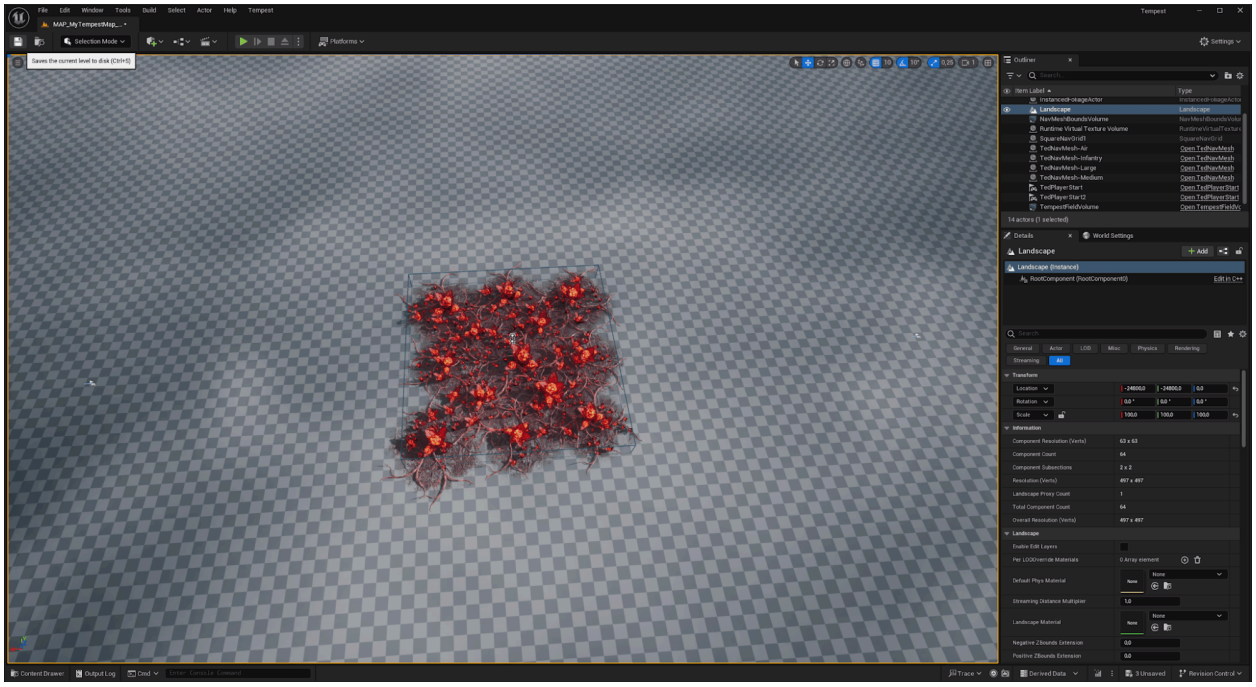
**NOTE**

*You can edit (or replace) your Mini-Map texture with another texture. This means you can customize your Mini-Map image with other tools such as photoshop, or paint, if you'd like!*

Additionally, if you look in your **MapInfo** asset, you'll see the Mini-Map image was also automatically added here:




To make sure your Mini-Map is saved, **>CLICK<** on 'Save' in the top left, or CTRL+S:

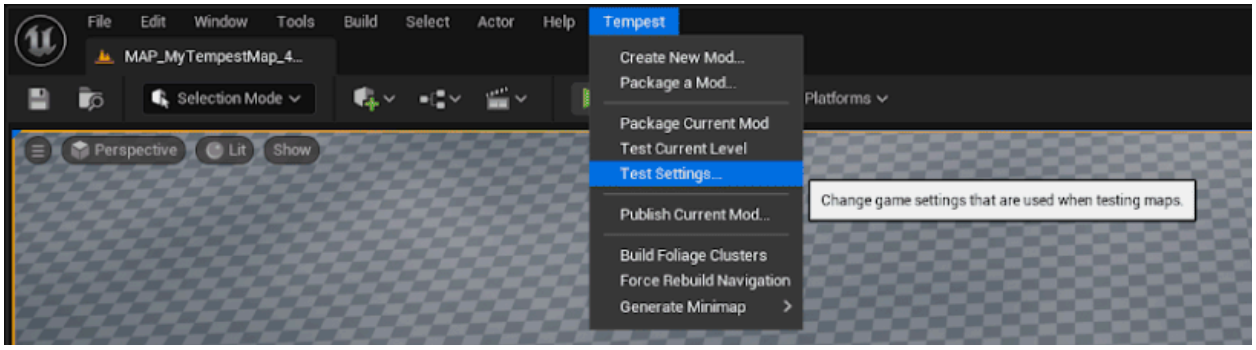


You may also want to >CLICK< on ‘File’ in the top left, and >CLICK< on ‘Save All’ (Shortcut: CTRL + SHIFT + S). This saves all assets you’ve recently changed.

## Test Current Level

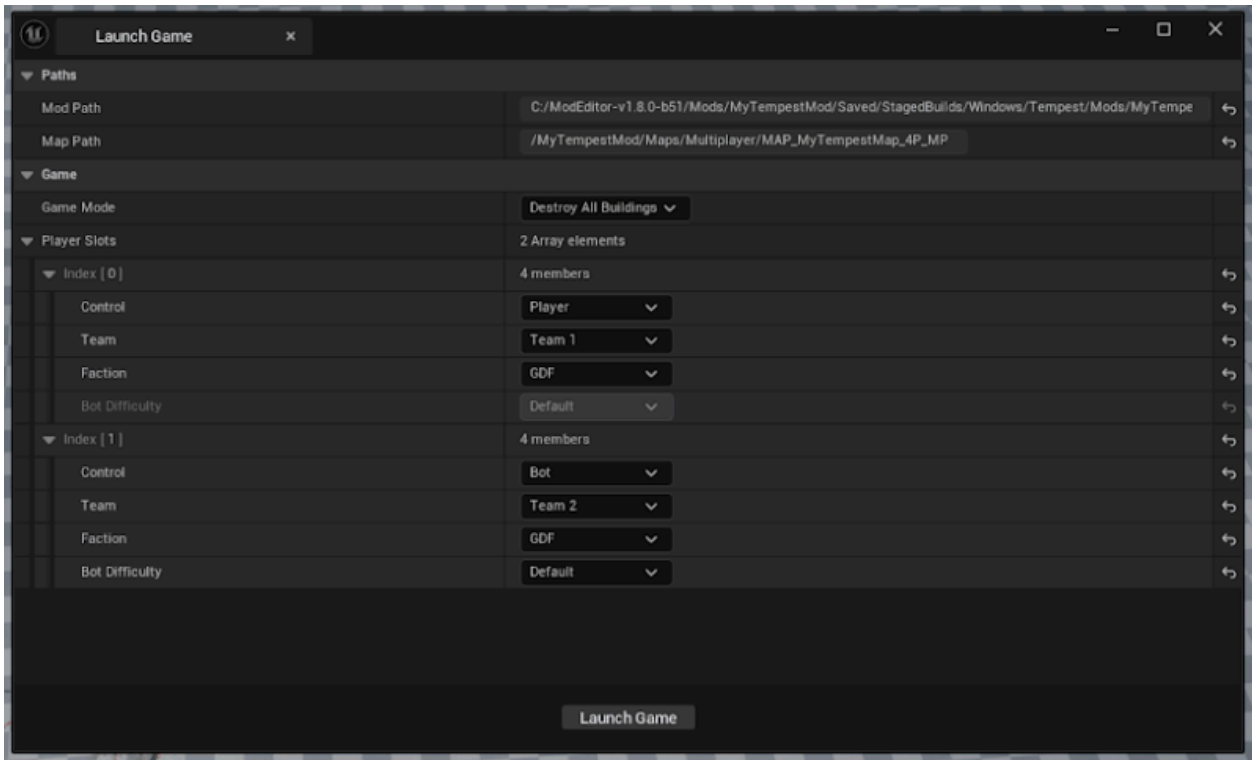
Now that your **MapInfo** has been filled, you should be ready to test your level inside Tempest Rising. To do this,

First, >CLICK< on the ‘Tempest’ menu in the Top Navigation Bar. Then, >CLICK< on ‘Test Settings…’:



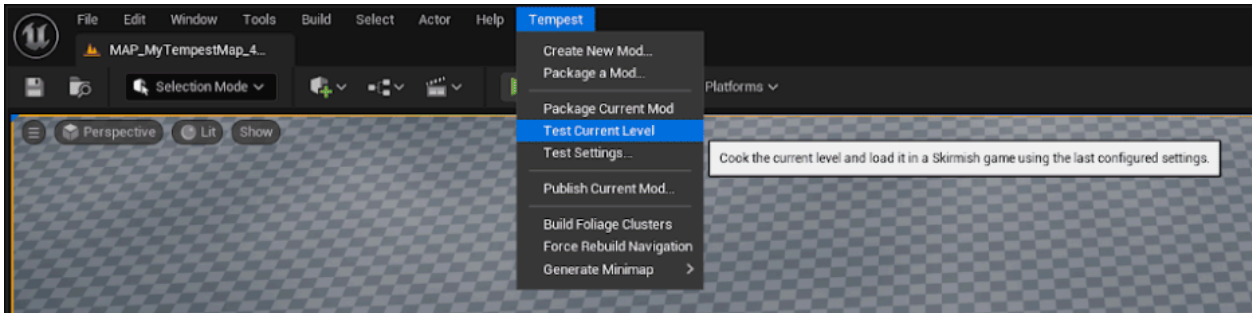

This is where you can configure your Test Settings. For example, which Game Mode / Win Condition is applied when you ‘Test Current Level’, and which opponents you have.

In the example below, I've **>CLICKED<** on Player Slots to fold out Index 1 (Player2), which I've **>SET<** to be a GDF Normal Bot on Team 2:




When you're ready, head back out to the 'Tempest' top menu, and **>CLICK<** on 'Test Current Level'.

This will automatically package your game, and open up Tempest Rising with your level for testing.

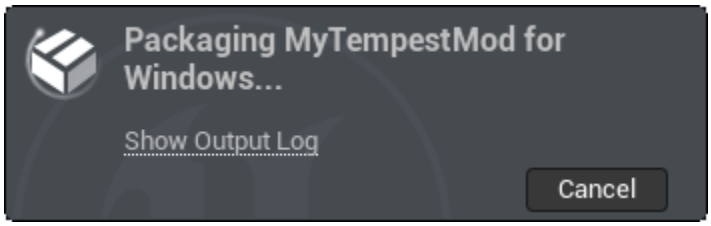


**NOTE**

*You have to specify your Tempest Rising install path the first time you do this. Likely this will be in your steamapps folder. The default path is something like:*

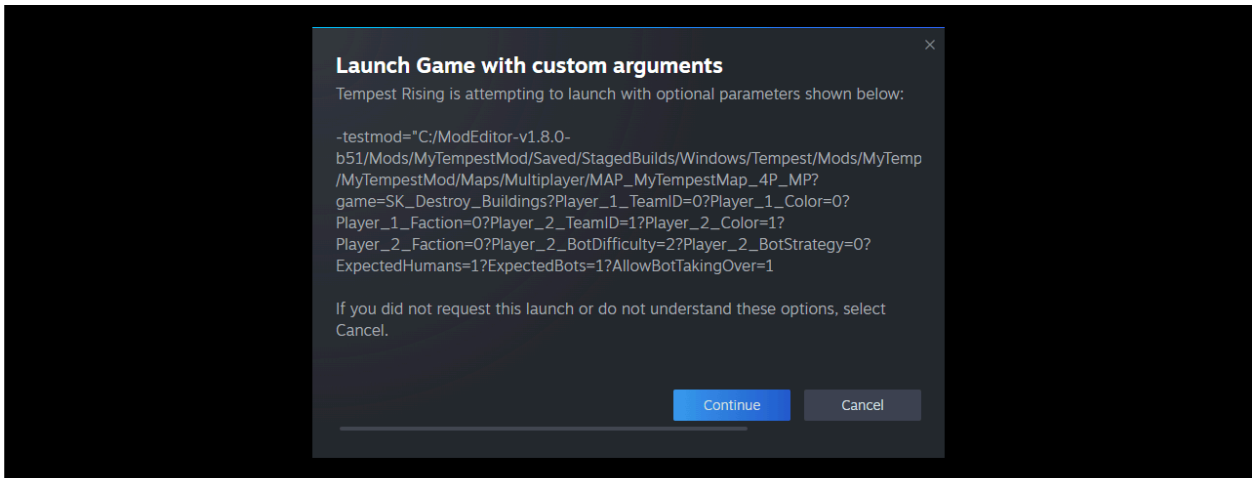
*C:\Program Files (x86)\Steam\steamapps\common\Tempest Rising*

After you clicked on 'Test Current Level', the following spinner should appear in the bottom right corner of your viewport. Packaging may take a few minutes, depending on your PC:

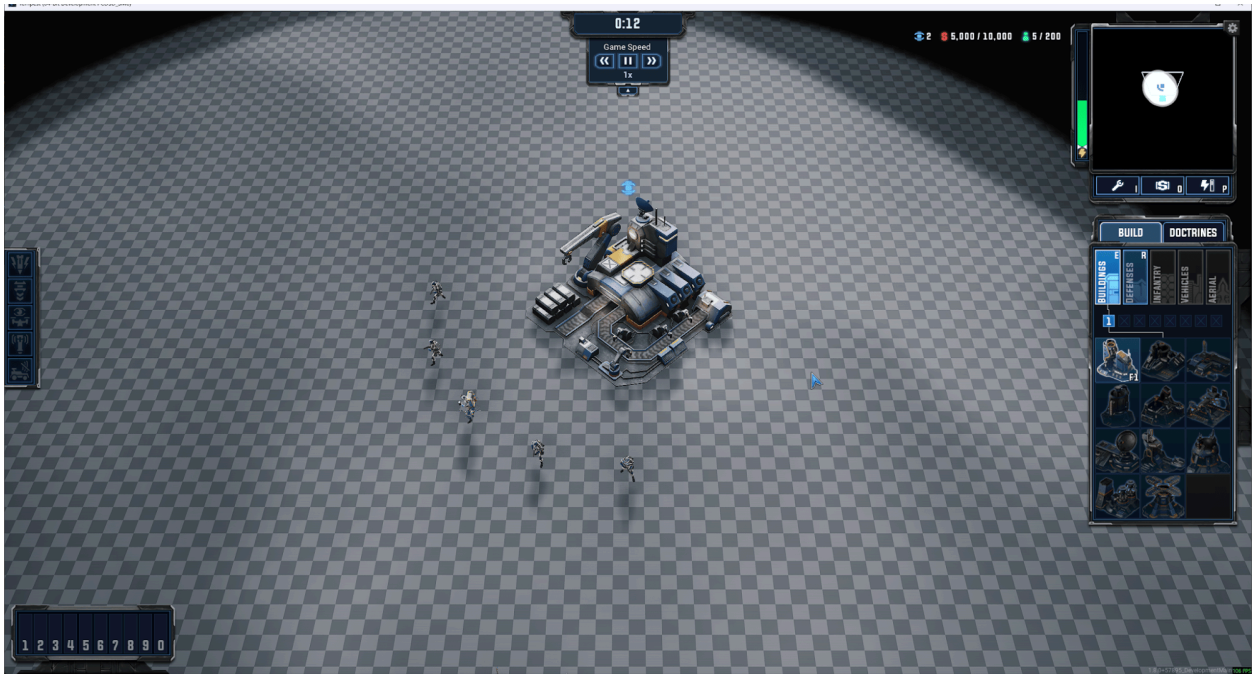



Once you're done packaging, Steam will automatically open (provided you have Steam installed, and own Tempest Rising).

The following prompt should now appear in your Steam window. **>CLICK<** on 'Continue':




You should now be in-game, in your Tempest Rising map:

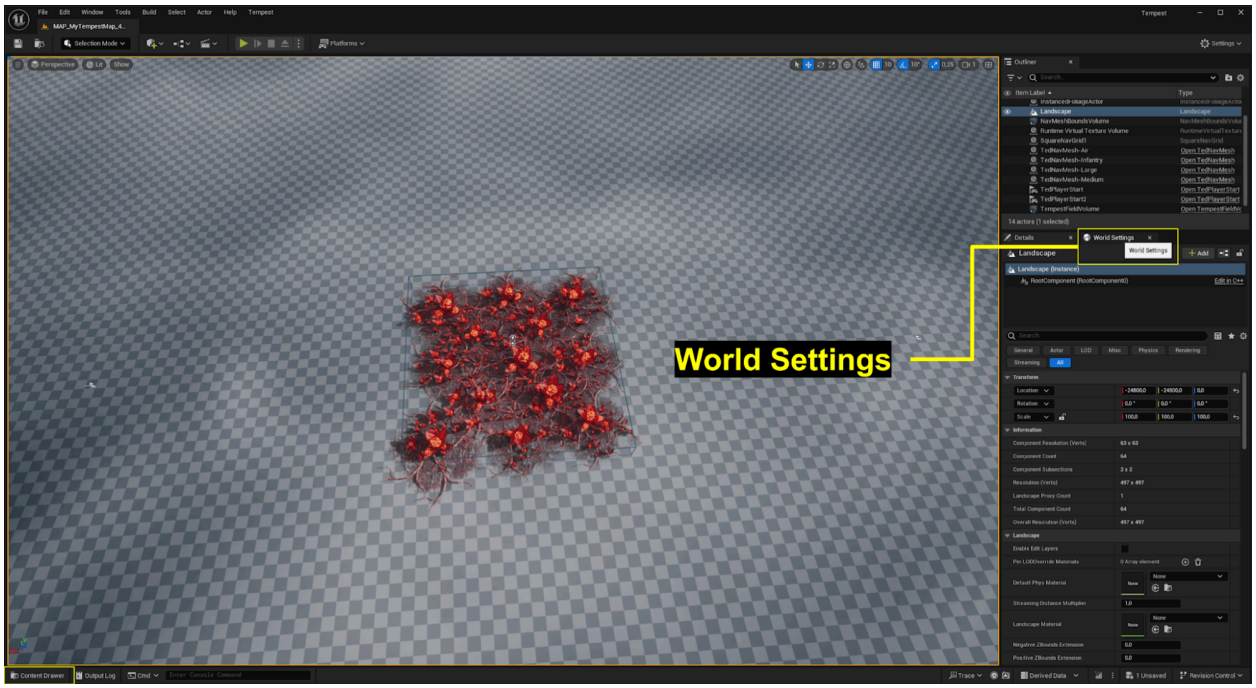



Additionally, if you set up a bot opponent in Slot 2 (Index 1), it should be preparing to fight in the Fog of War:

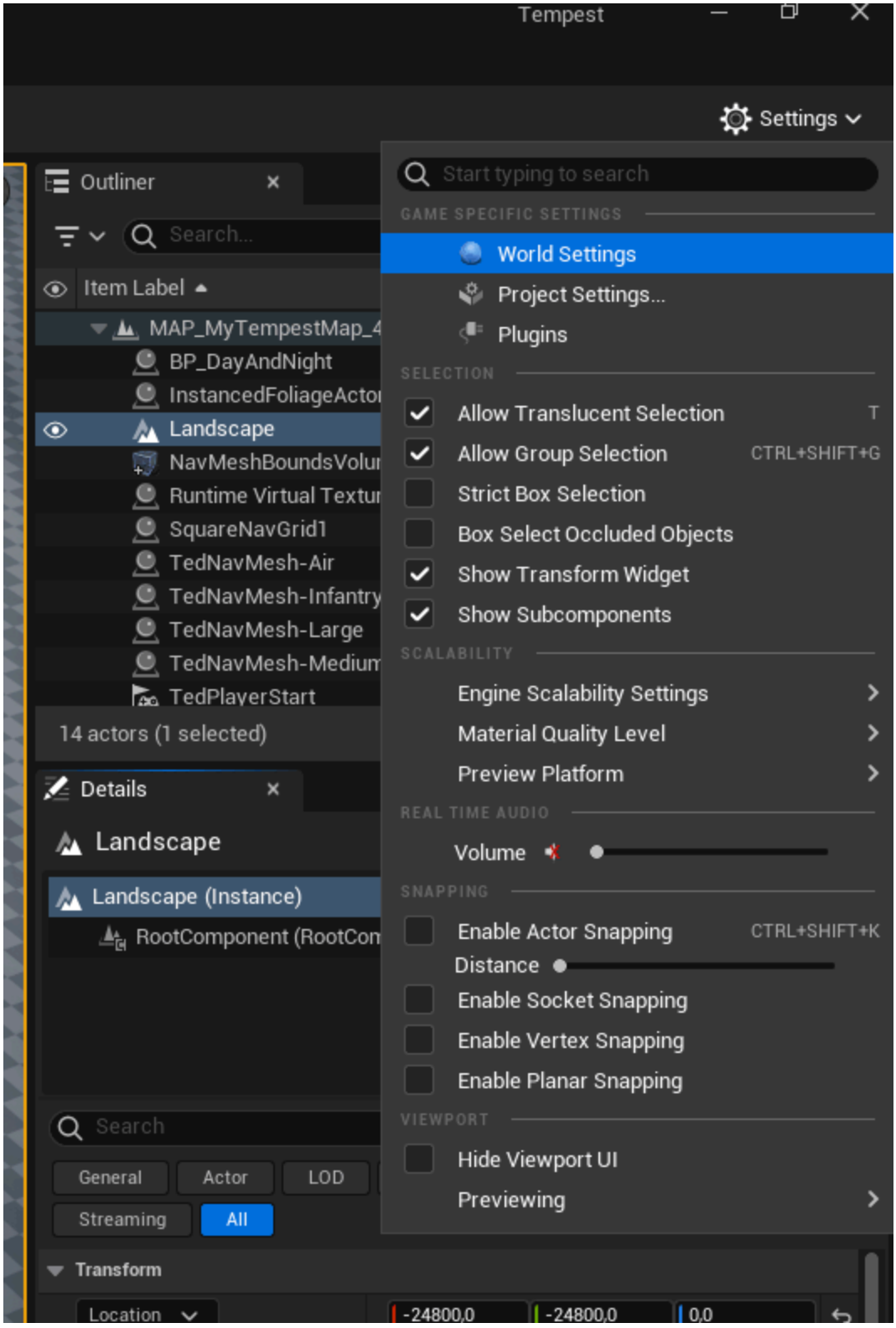
## Configure World Settings

As you may have noticed, the Fog of War is pitch-black in the example above. This is the default Fog of War behavior for the Singleplayer Campaign - **NOT** for MP / Skirmish maps, where Terrain is always Visible, even through Fog of War.

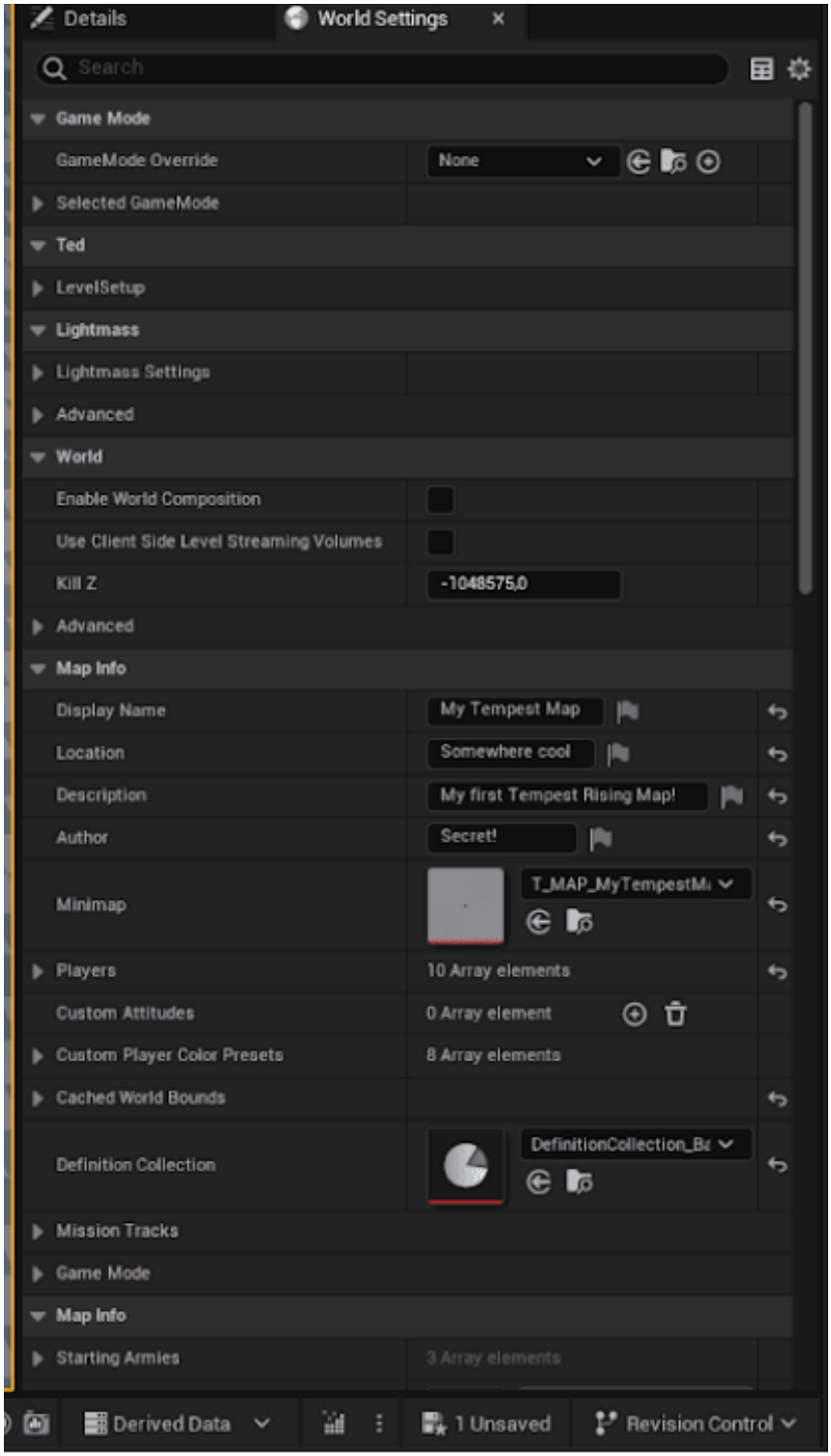
To configure this, find the **World Settings** tab, and **>CLICK<** on it:




**REMINDER: IF YOU CANNOT** see the World Settings panel, you will have to **>CLICK<** on 'Settings' in the top left of the Unreal view, and enable the World Settings panel there first:



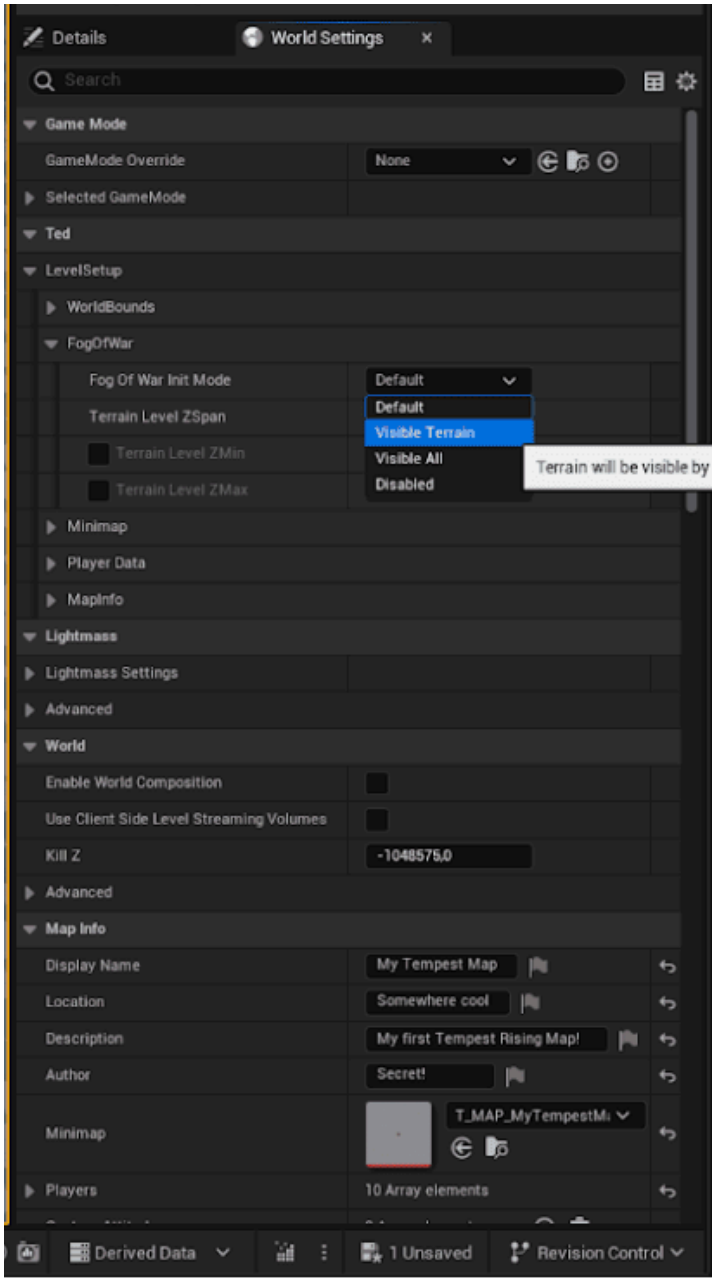

When you then **>CLICK<** on the World Settings panel to the right in the Unreal view, this is the first view you will see:


Details
World Settings
Search
Game Mode
GameMode Override
None
Selected GameMode
Ted
LevelSetup
Lightmass
Lightmass Settings
Advanced
World
Enable World Composition
Use Client Side Level Streaming Volumes
Kill Z
-1048575.0
Advanced
Map Info
Display Name
My Tempest Map
Location
Somewhere cool
Description
My first Tempest Rising Map!
Author
Secret!
Minimap
T_MAP_MyTempestMi
Players
10 Array elements
Custom Attitudes
0 Array element
Custom Player Color Presets
8 Array elements
Cached World Bounds
Definition Collection
DefinitionCollection_Ba
Mission Tracks
Game Mode
Map Info
Starting Armies
3 Array elements
Derived Data
1 Unsaved
Revision Control

**>CLICK<** on ‘LevelSetup’ to fold it out, same with ‘FogOfWar’. Here, you’ll see a Drop-Down Init Mode.

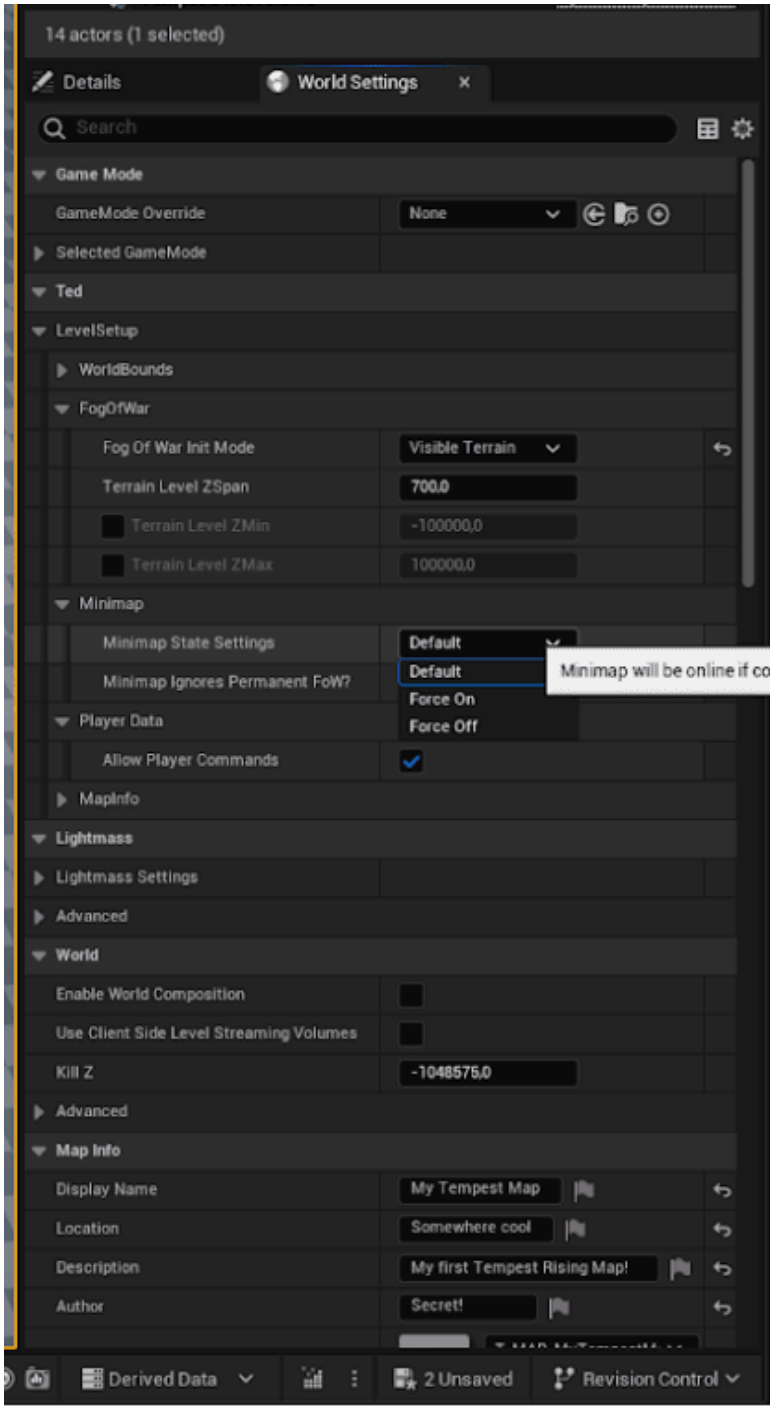
**>CLICK<** on it, and select ‘Visible Terrain’.

This is the default setting for Tempest Rising MP & Skirmish Maps. You can also fully disable the Fog of War, if you so desire.



Additionally, you can >CLICK< on and fold out the ‘Minimap’ section. Here, in the Minimap State Settings, you can force the Minimap On, or Off.

Additionally, under Player Data, you can tick ‘Allow Player commands’. This is what determines if players can use Support Powers, or not, in a map.



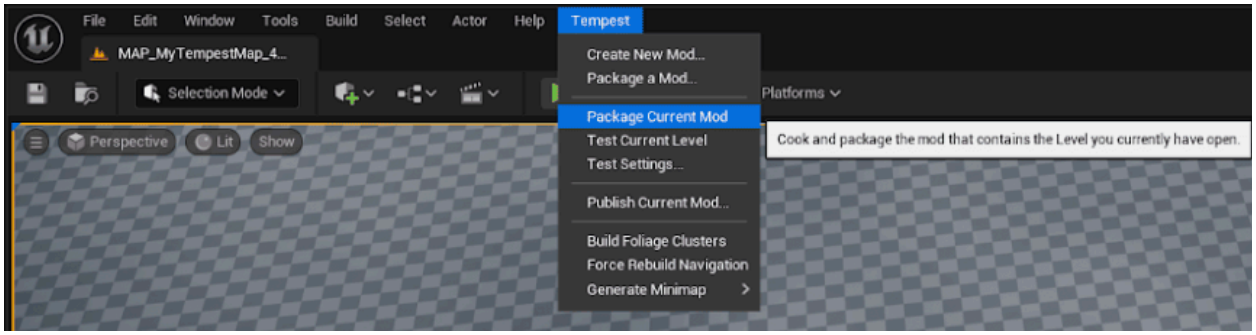
After you've set the Fog of War to 'Visible Terrain', head back to the Tempest top menu and **>CLICK<** on 'Test Current Level' again.

This time, you should see the terrain through the Fog of War:

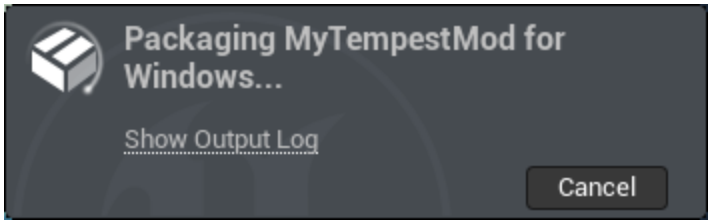

## Packaging Your Mod

You are now ready to package your mod! It's similar to 'Test Current Level' - main exception being that you get a Mod Folder, that you can import into Tempest Rising, or share with your friends for online play.

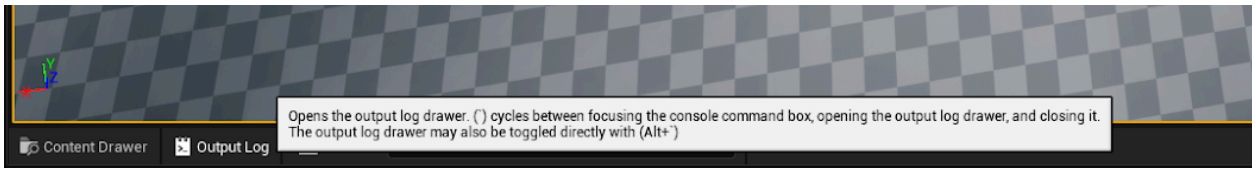
To package your mod, open the top 'Tempest' menu, and **>CLICK<** on 'Package Current Mod':



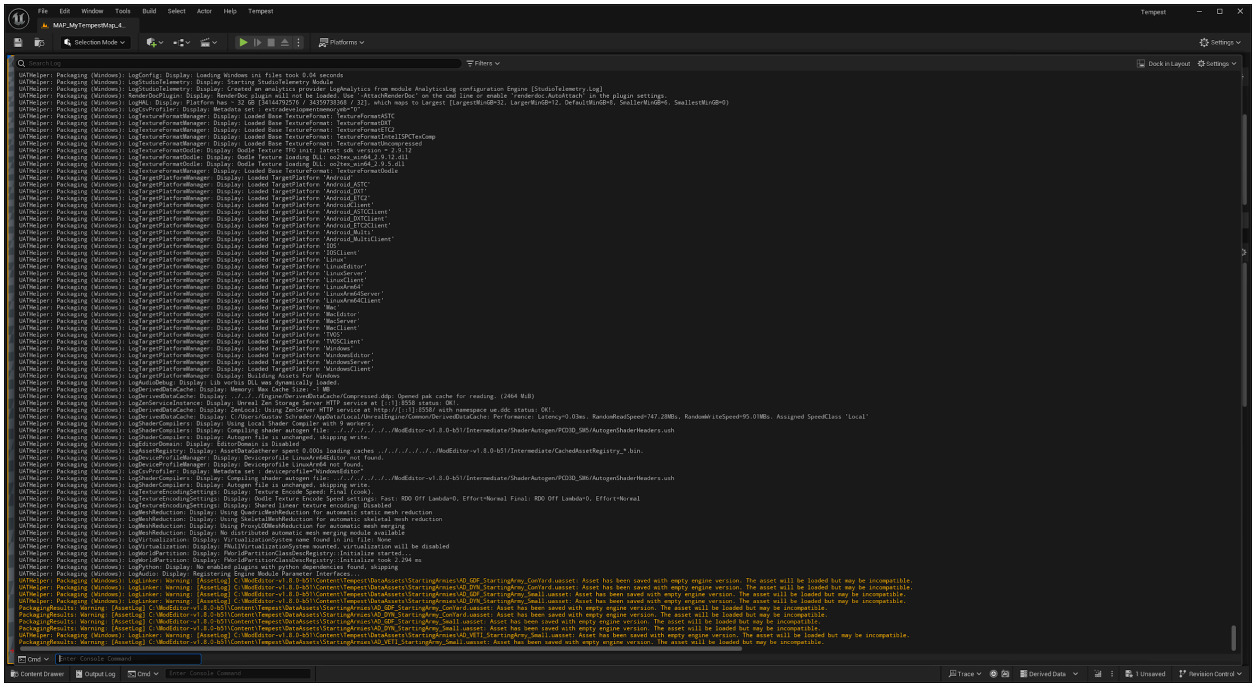

You should now see the 'Packaging' status message in the bottom right of your viewport:



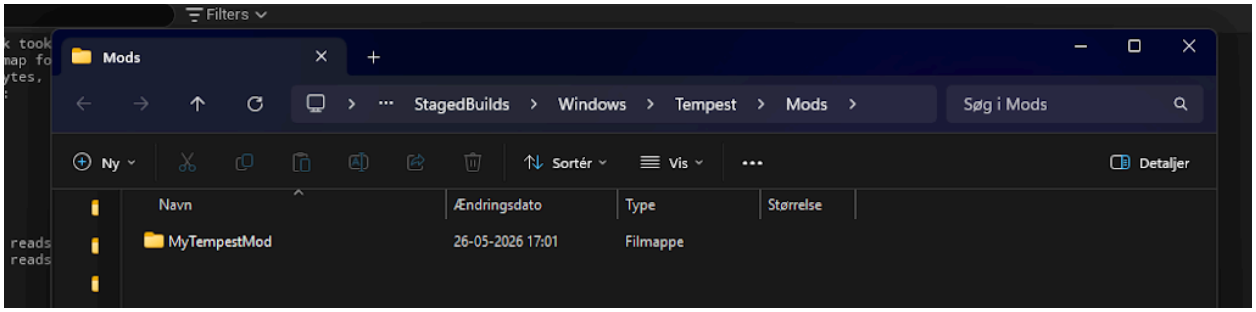

If you want, you can also open the ‘Output Log’ in the bottom left, to see the on-going packaging process:




The packaging process looks like this:



Once your Mod is done packaging, you’ll get a pop-up windows folder, with your Mod folder in it:




Congratulations! You've packaged your first mod for Tempest Rising!

## Playing Your Mod

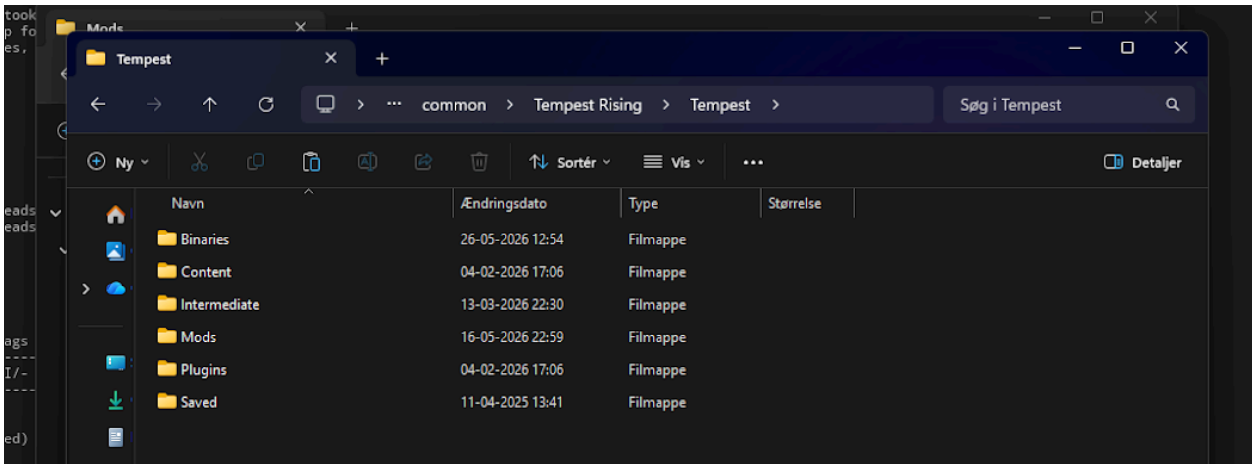
In order to play your Mod, you'll need to import it into Tempest Rising, and select it in either Skirmish (Offline Play), or Custom Games (Online Play).

First, **>COPY<** your Mod Folder from the packaging you did earlier.

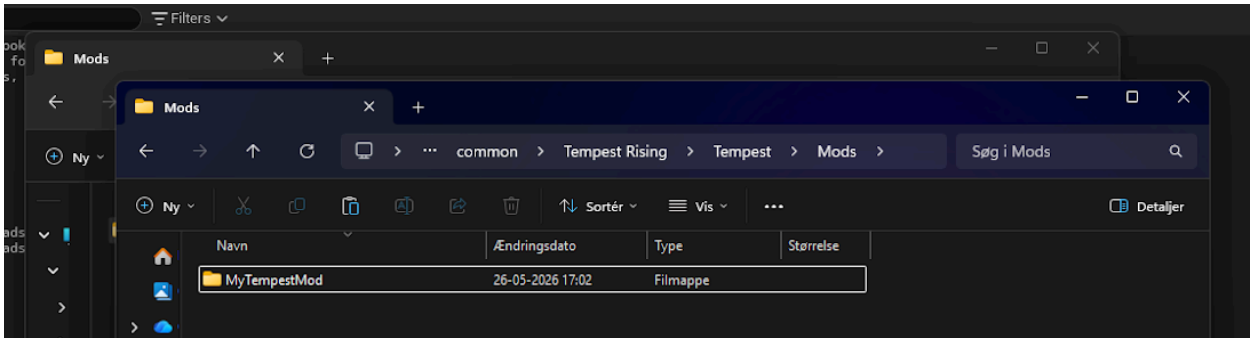
Then, find your installation folder for Tempest Rising, and **>OPEN<** the Tempest Folder here. Typical path would be:

*C:\Program Files (x86)\Steam\steamapps\common\Tempest Rising\Tempest*

Here, **>OPEN<** the 'Mods' folder:




**>PLACE / PASTE<** your Mod folder inside the 'Mods' folder:




Next, **>OPEN<** Tempest Rising:

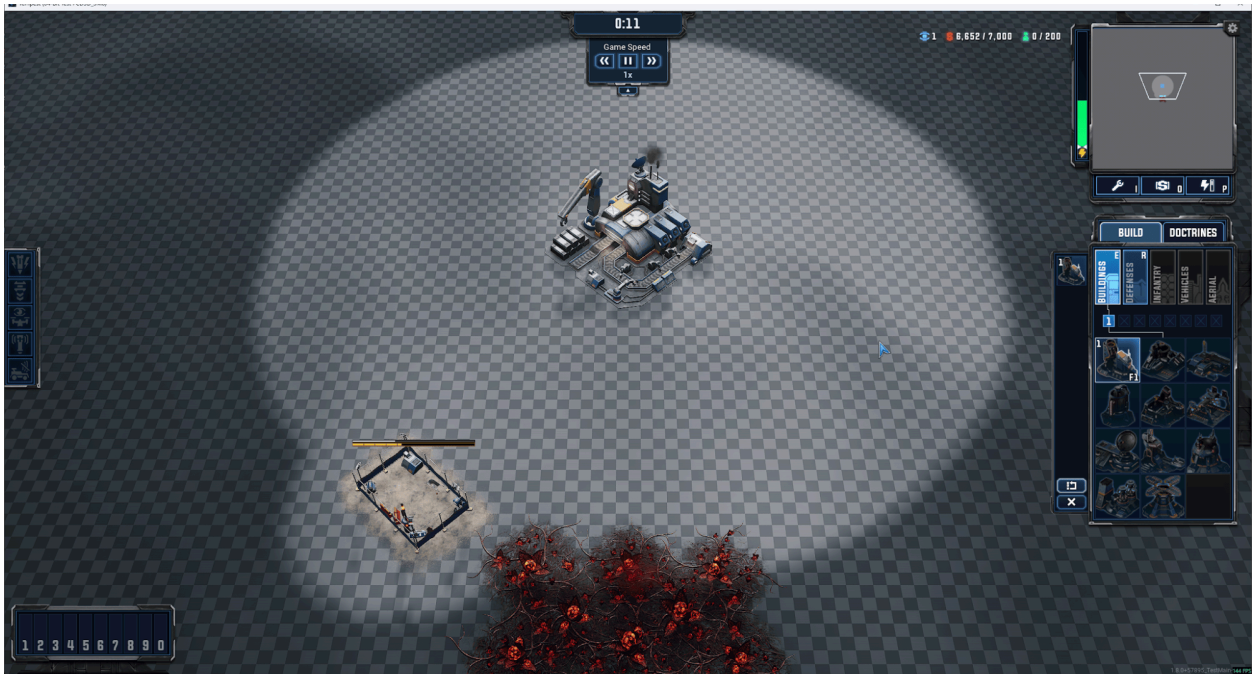

From the main menu, **>OPEN<** Skirmish:


**>CLICK<** on ‘Change Map’. This will bring up the ‘Select Map’ UI:

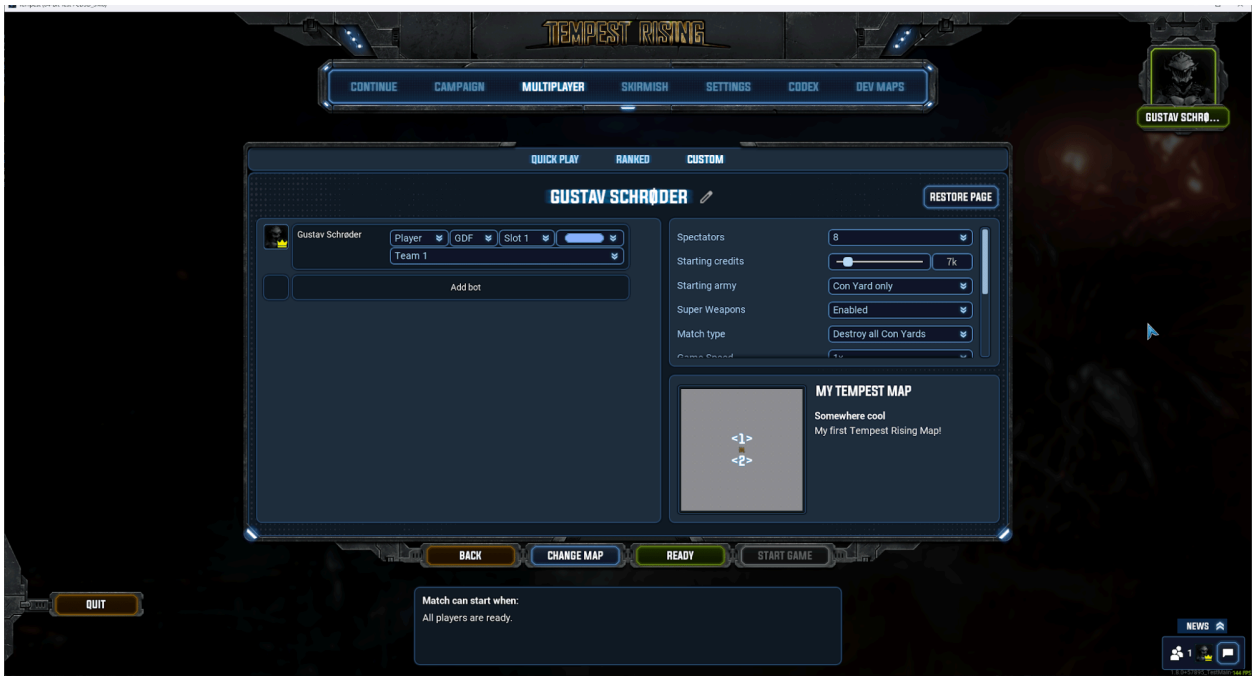

**>SCROLL<** until you find your Custom Map. **>CLICK<** on it to select it, and **>CLICK<** on ‘Confirm’:


You now have your Custom Map selected for a Skirmish match. In the example below, I've added a normal GDF bot to play against:


Going into the game, you are now able to play your Custom Map inside Tempest Rising.




The same exact flow applies to Multiplayer (Custom Games). Simply host a lobby as you normally do, and select your Custom Map from the ‘Change Map’ menu.




## NOTE

*Both you and your friends must have the same Custom Map installed to play maps together.*

*Your friend will get a warning pop-up if he does not have your Map installed, prompting him to download it from Steam Workshop (if it's uploaded there).*

## Sharing Your Mod

In order to share your Mod/Maps, you can do two things:

A) You can upload your Mod to Steam Workshop (see main documentation for details)
B) You can send a .zip file to your friend directly, and have them place it into their Mods folder inside their Tempest Rising installation folder.

## Further Recommendations

That's about it for the very basics of creating a Tempest Rising map.

Though there is a lot more to creating a fully fledged map - we recommend you read the main Tempest Rising Map Editor documentation for further Tempest specific information.

For further information on Unreal Engine, we recommend the 'Getting Started' guide for Unreal Engine 5: https://dev.epicgames.com/community/unreal-engine/getting-started

# A Note on Fab Marketplace

Unreal Engine has integration with Fab, via the Epic Games Launcher. Fab is an asset marketplace, where you can either download free assets, or purchase assets, for easy import into Unreal Engine projects.

You can import assets into the Tempest Rising Map Editor project as well, though with an extra necessary step, since your Mod Content Folder is not located in the default Content folder, and instead under Plugins.

In this next section, we will walk you through how to import assets from Fab, and include them in your Tempest Rising Map.

**WARNING**

*Since assets from Fab are not made by Slipgate Ironworks, we cannot ensure their stability or quality.*

*You are also responsible for not breaking any licenses related to the assets you download (for example, some assets allow you to use them in packaged projects, but **NOT** share them as-is)*

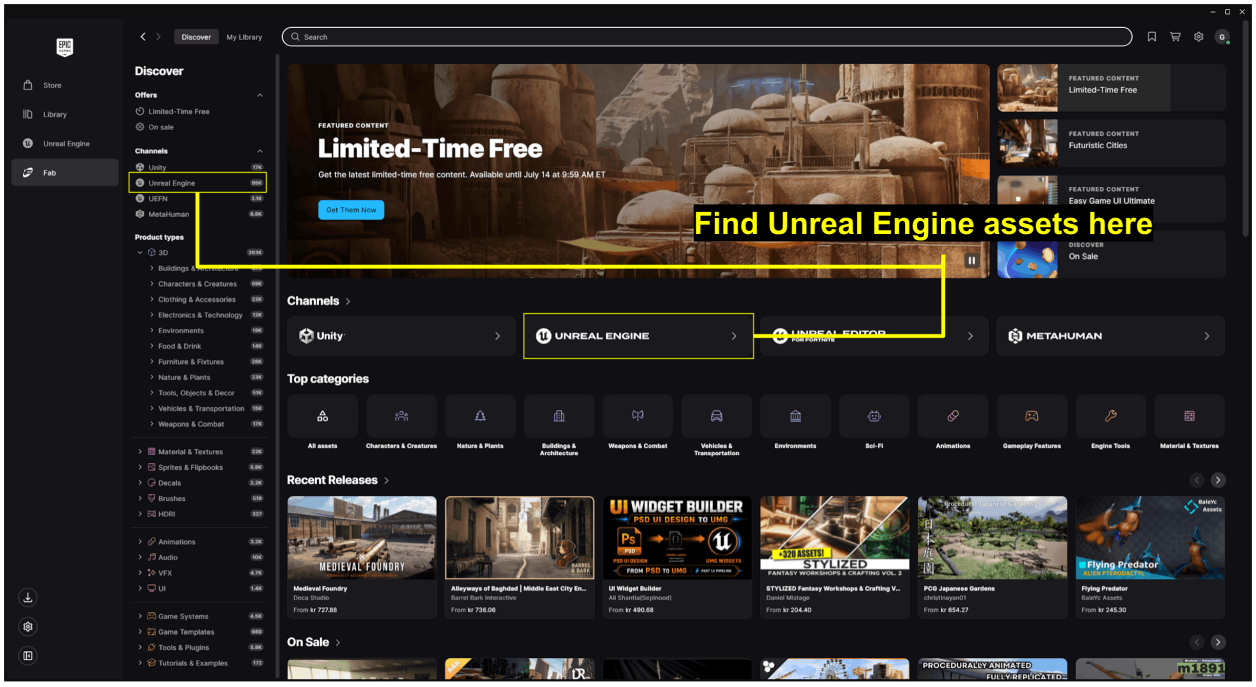
*Additionally, imported assets will significantly increase the filesize of your Map. It's best to examine the Props folder included with the Map Editor project first, and the **Environment Art Guide**, to see if your needs can be met with what you already have.*

*If not, then Fab marketplace assets might be what you need.*

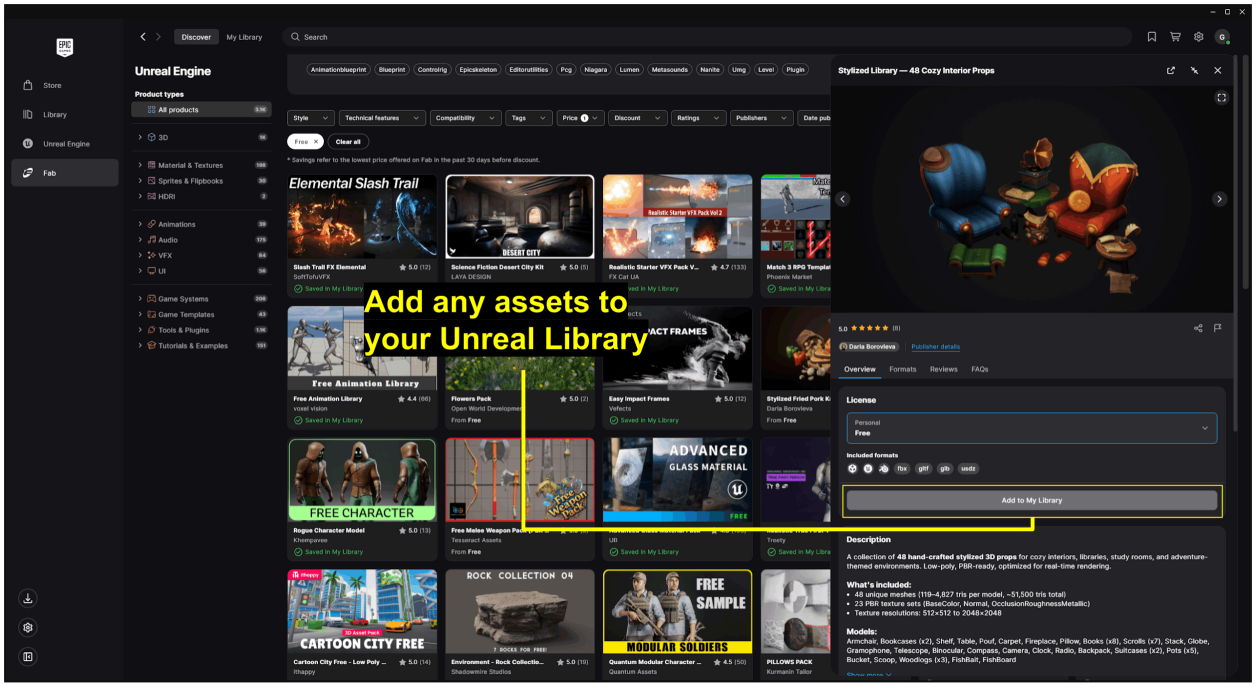
## First Time View (Fab)

Fab should be relatively intuitive to navigate; though the most important assets for you will be found under the "Unreal Engine" section.

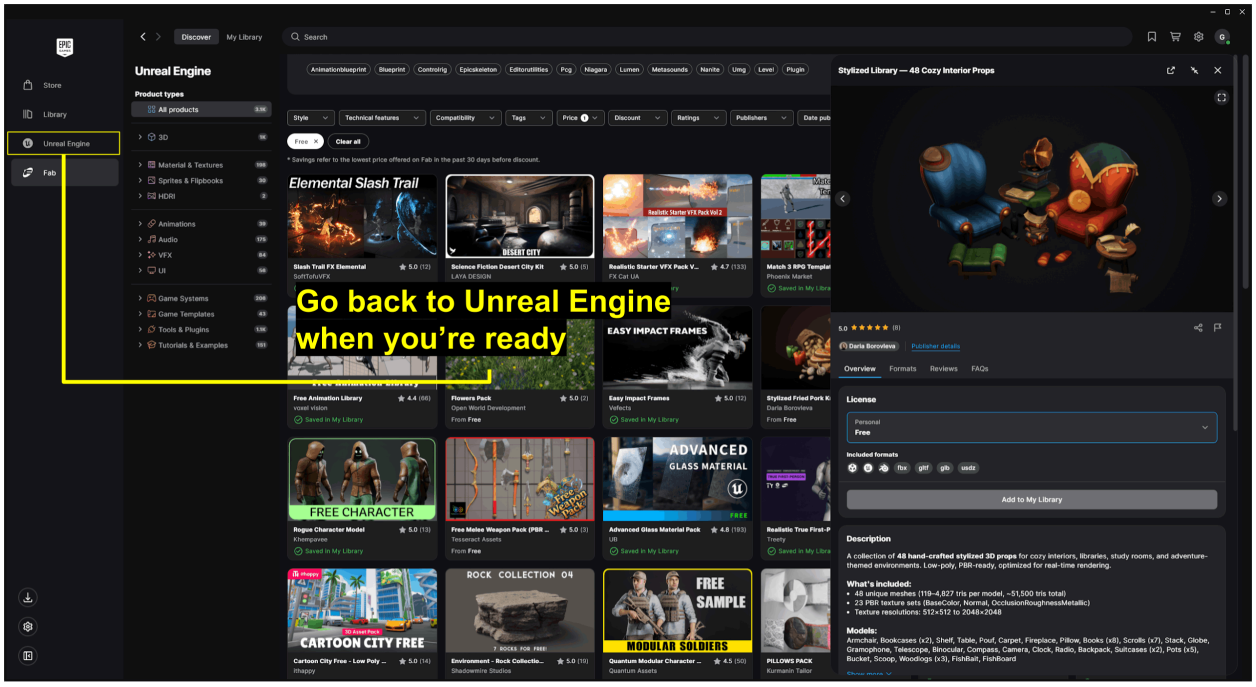
To go browse assets, **>CLICK<** on 'Unreal Engine' either to the left, or in the center navigation button.



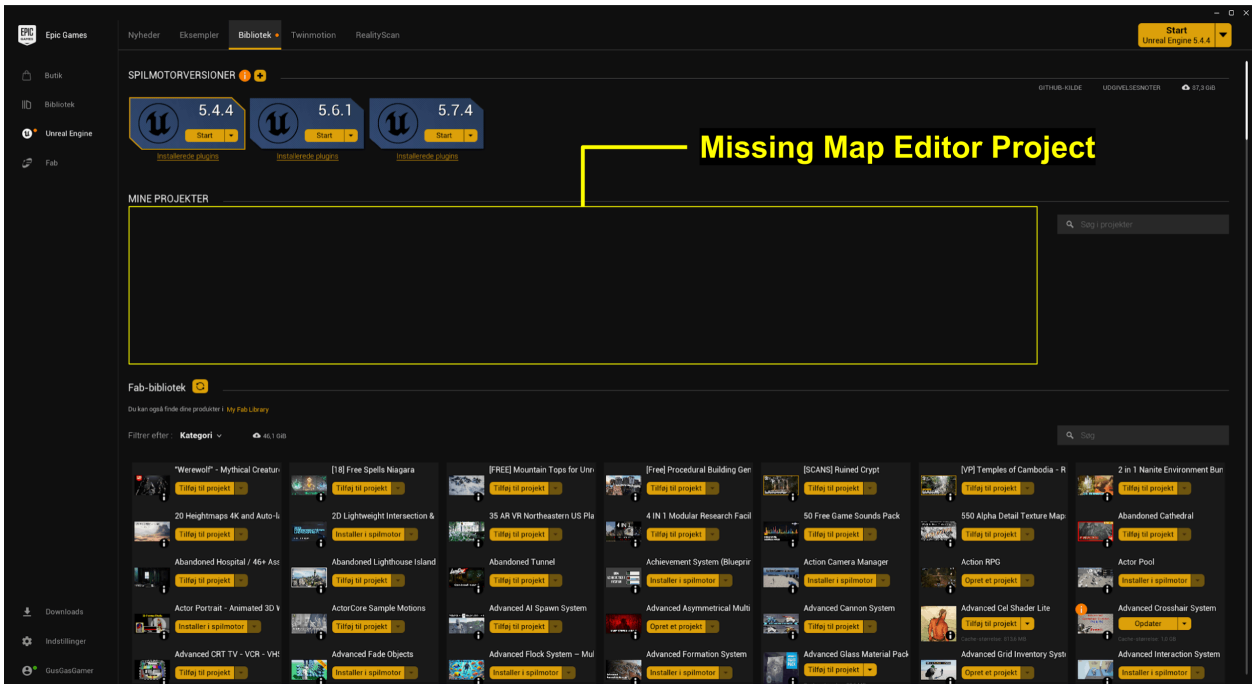

When you find an asset (or asset pack) that you want to use, you need to **>CLICK<** on 'Add to Library'. This will add the related asset pack to your Library, where you previously went to install Unreal Engine at the start of this guide.




When you are ready, **>CLICK<** on Library, and navigate back to the Library view:




In your **Library**, you can now add Fab assets directly to your projects. However, you likely cannot see your Tempest Map Editor project here, since it is not located in the Unreal Projects folder.



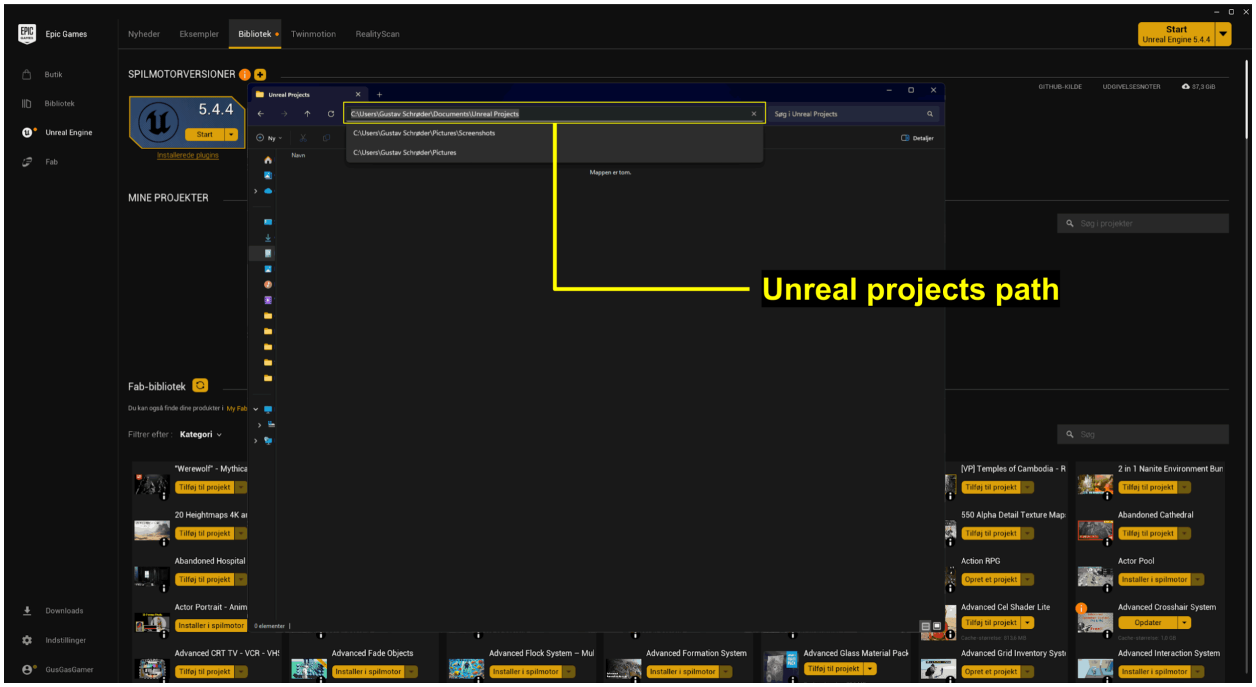

To add your Map Editor project to this view, you will have to move the Tempest Rising Map editor project into the default Unreal Engine projects path.

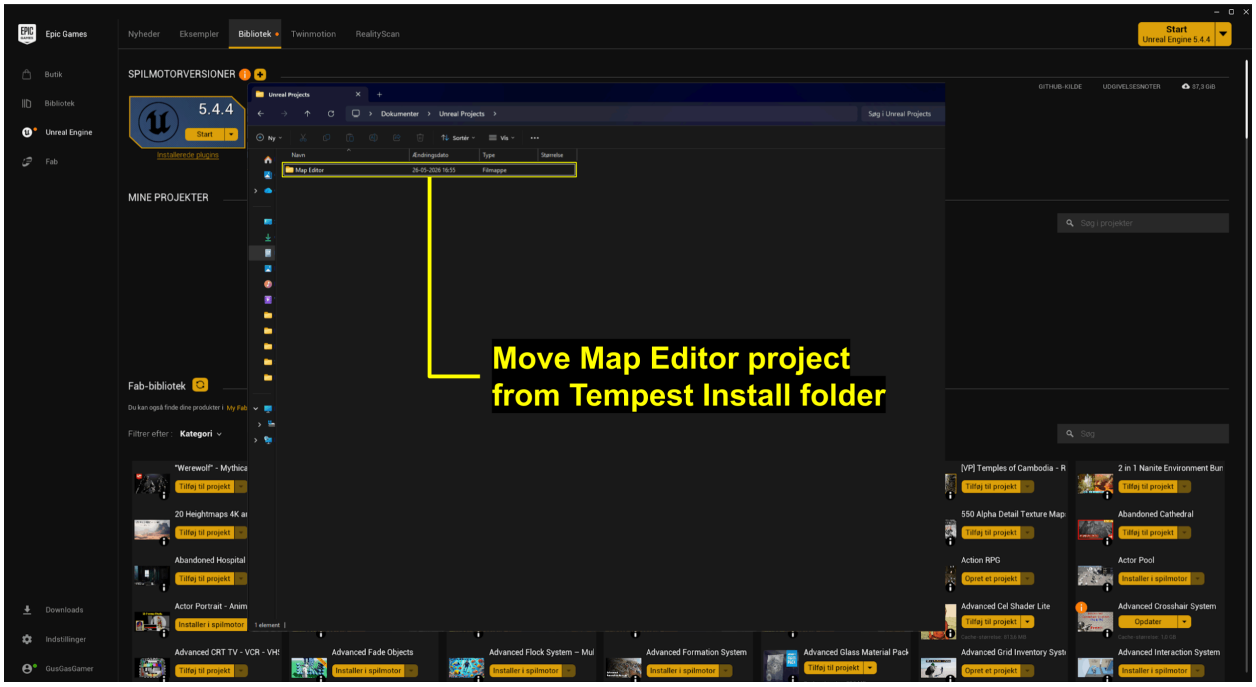
To do this, first **>FIND<** your Tempest Rising Map Editor Project folder.

Typical path for this will be:
**C:\Program Files (x86)\Steam\steamapps\common\Tempest Rising**

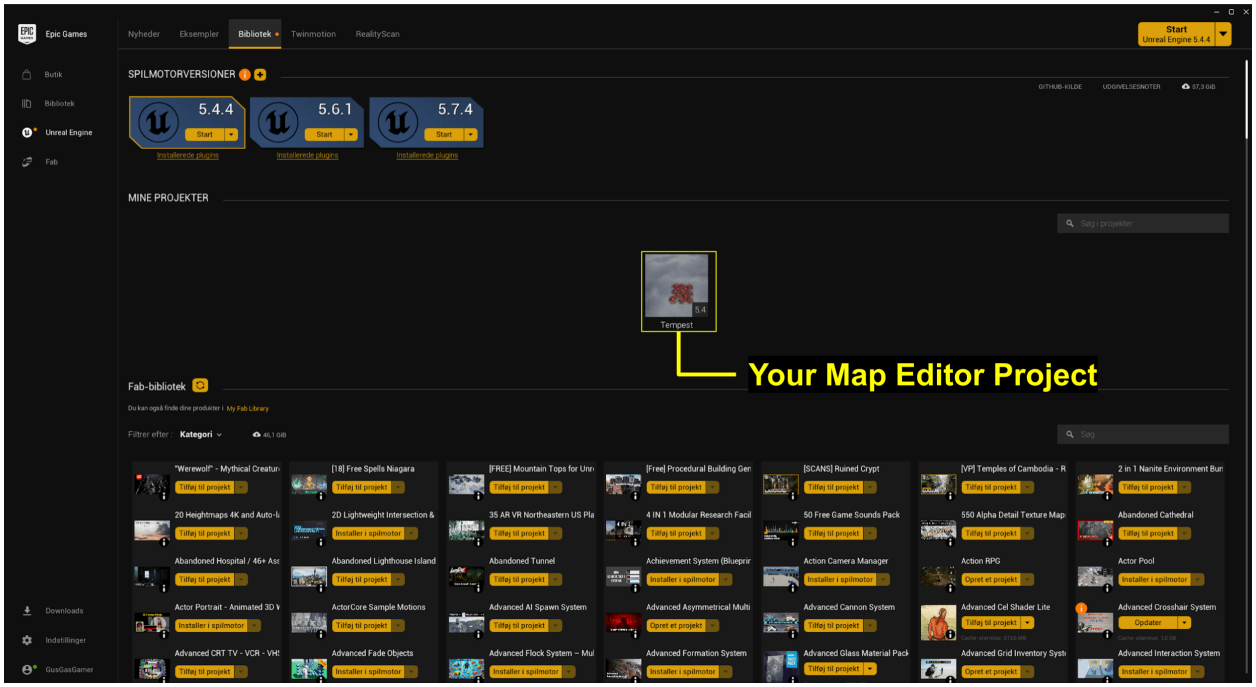
Once you've found the 'MapEditorProject' folder, you will need to **>MOVE<** it to the Unreal Engine projects folder.

Typical path for this will be:
**C:\Users\UserName\Documents\Unreal Projects**

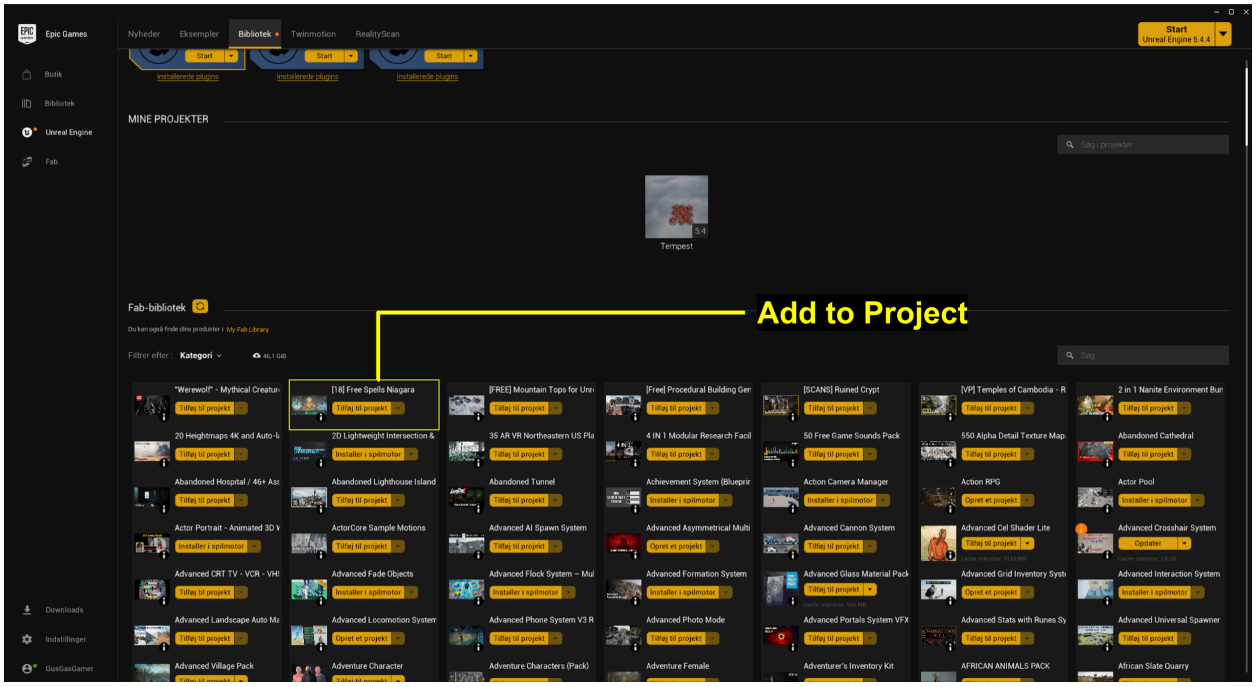




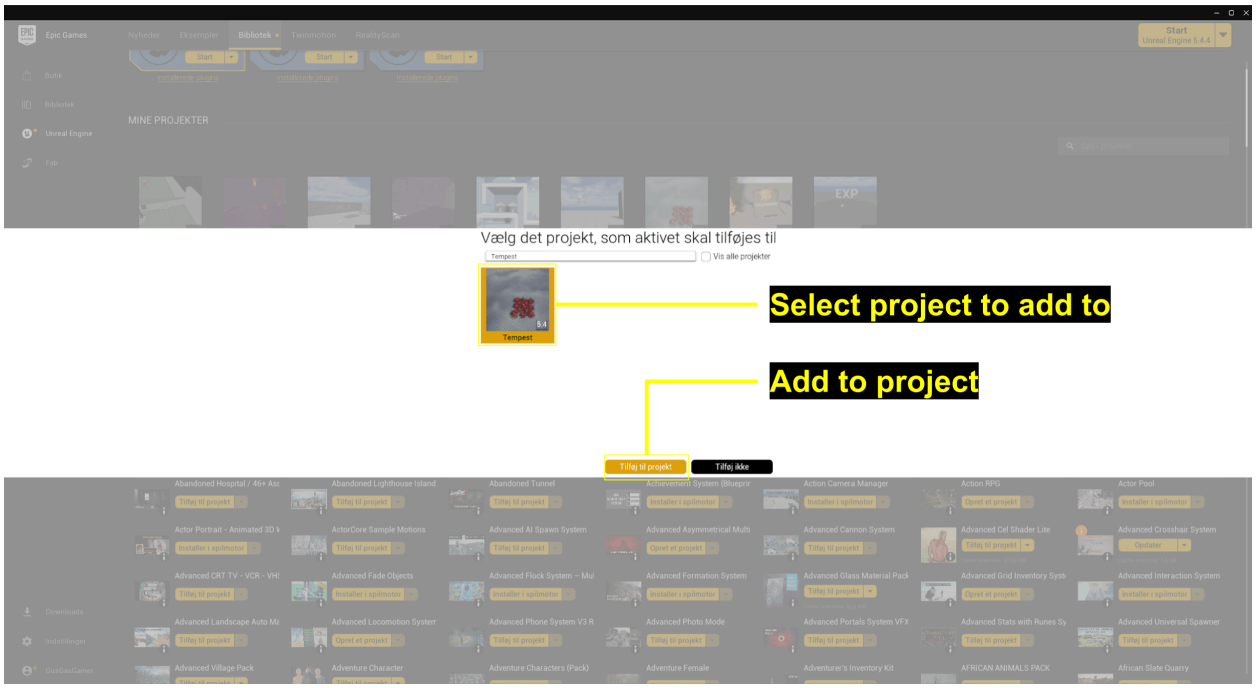

Your Map Editor project should now show up inside the Unreal Engine Library:



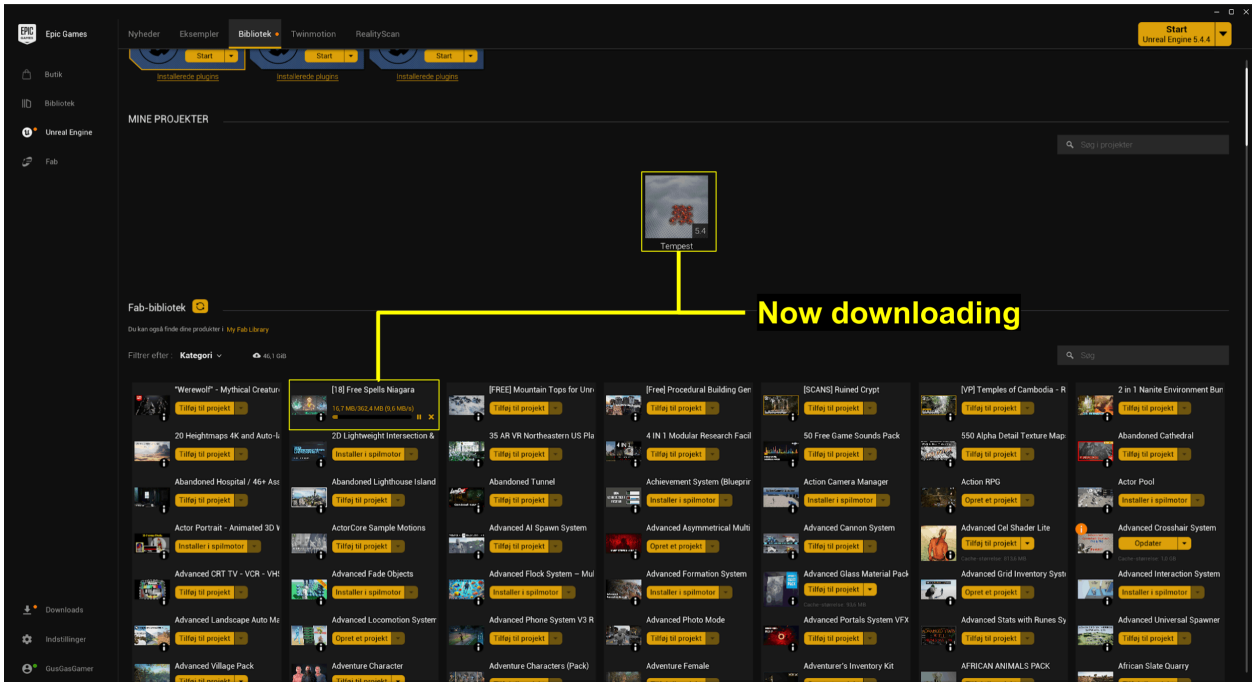

Next, you'll want to add one of your assets to the Map Editor Project. **>CLICK<** on 'Add to Project':




Next, you'll get a pop-up window asking which project to add the asset to. Select your Map Editor project, and **>CLICK<** on 'Add to Project':




You'll then see the asset pack begin downloading. Once it stops downloading, your asset pack is now inside the Map Editor Project:

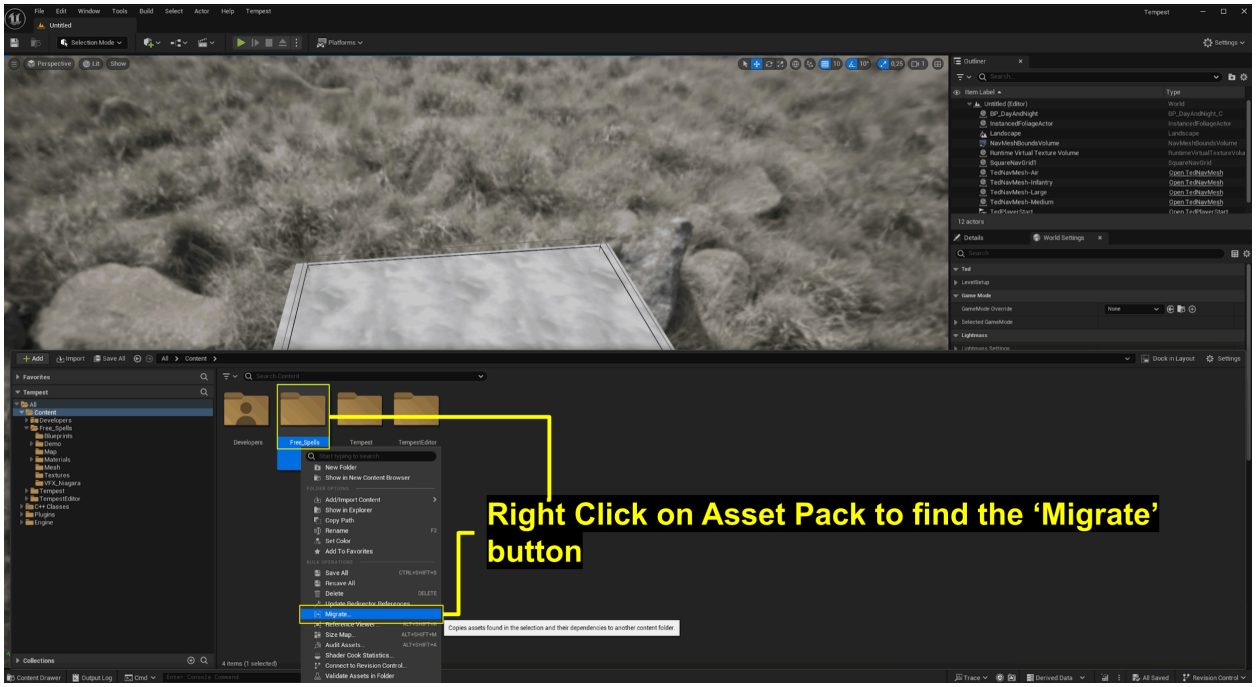



Once it’s done downloading, **>CLICK<** on the Map Editor project to open it.

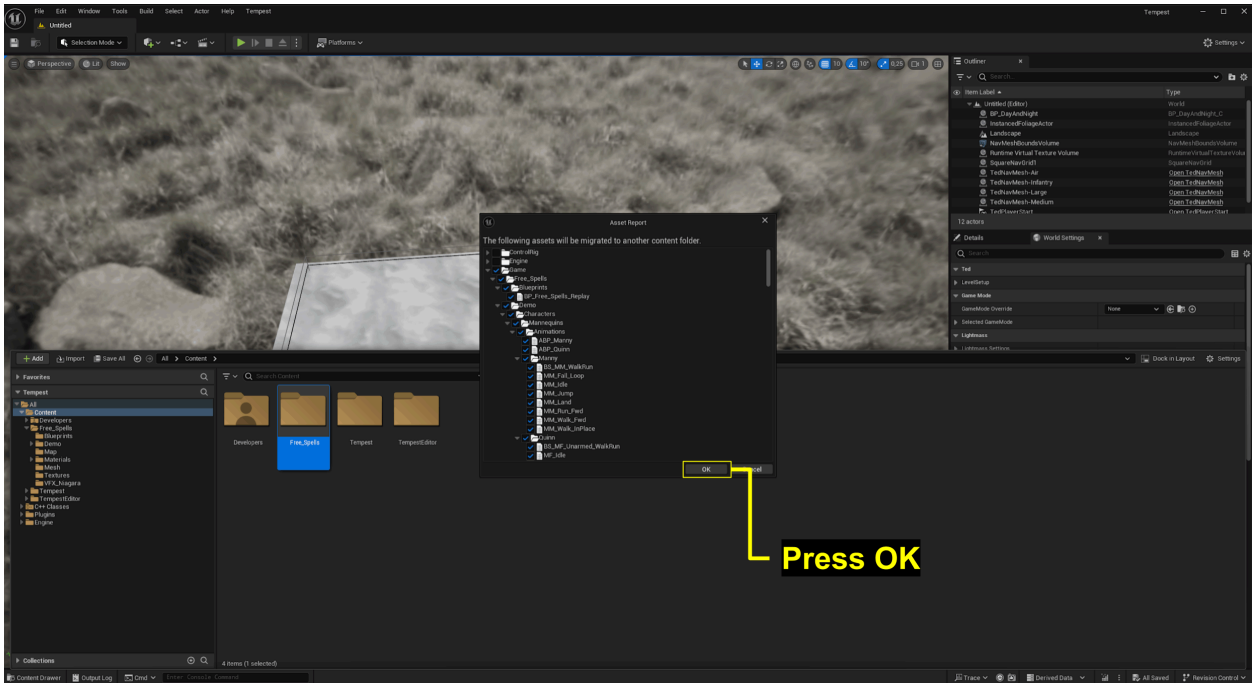
We will want to **Migrate** the asset pack, as it is in the wrong Content folder by default.


To fix this, first **>RIGHT-CLICK<** on the Asset pack (Free_Spells in this example).

In the drop-down that shows up, then find the **Migrate** button, and **>CLICK<** on it.



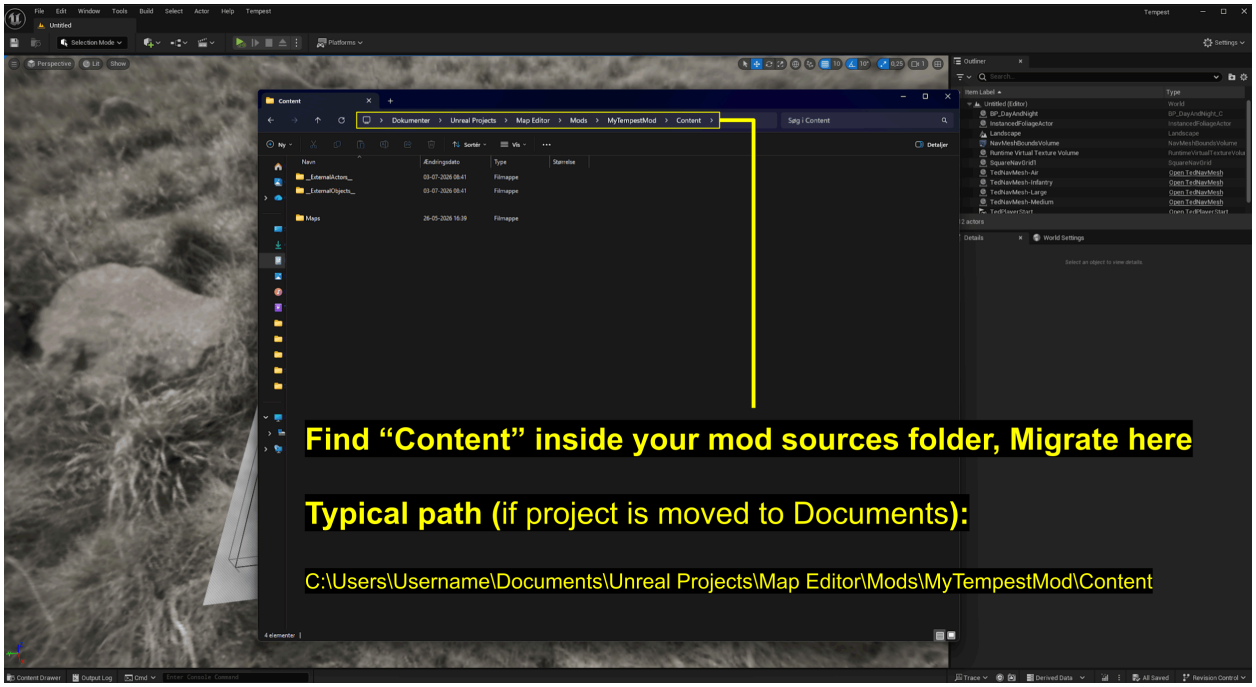

You will get the following prompt, asking which assets should be Migrated. Since the migration tool gathers all relevant dependencies, you can just **>CLICK<** on ‘OK’:



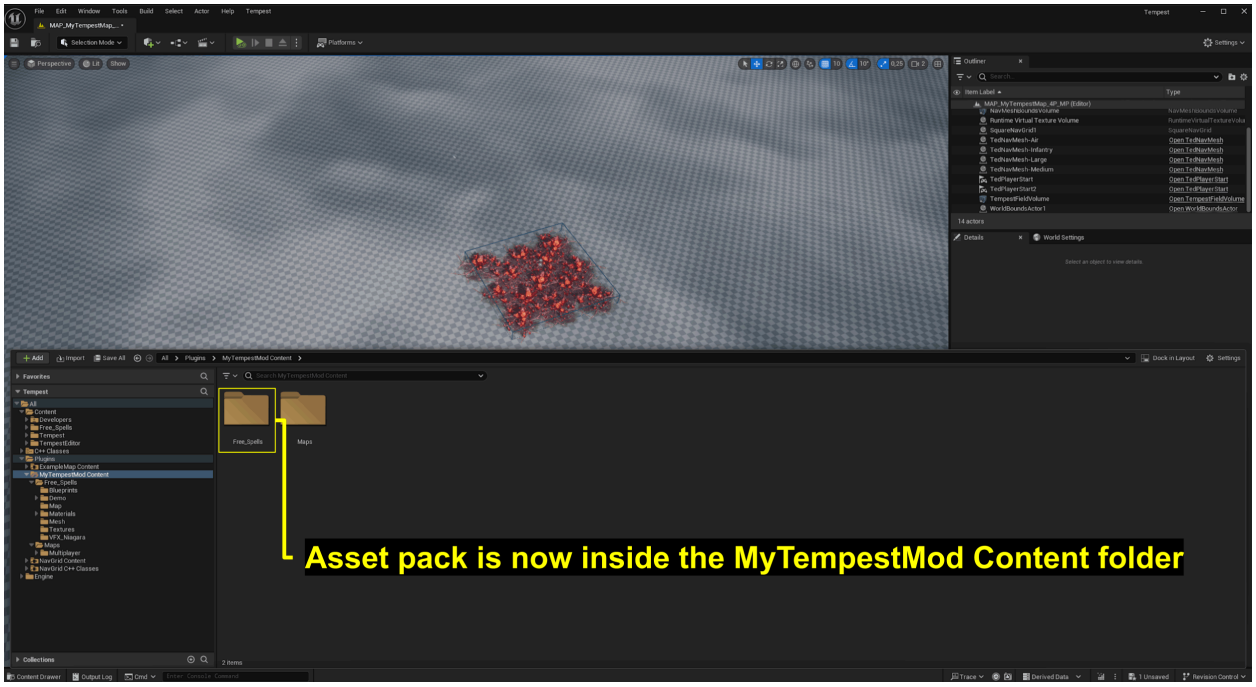
Next, you'll have to find the Content folder that you are migrating to. This will be your **Mod Content Folder**. It is the same folder where your Map files are located.

**>MIGRATE<** here once you've located the Mod Content Folder.

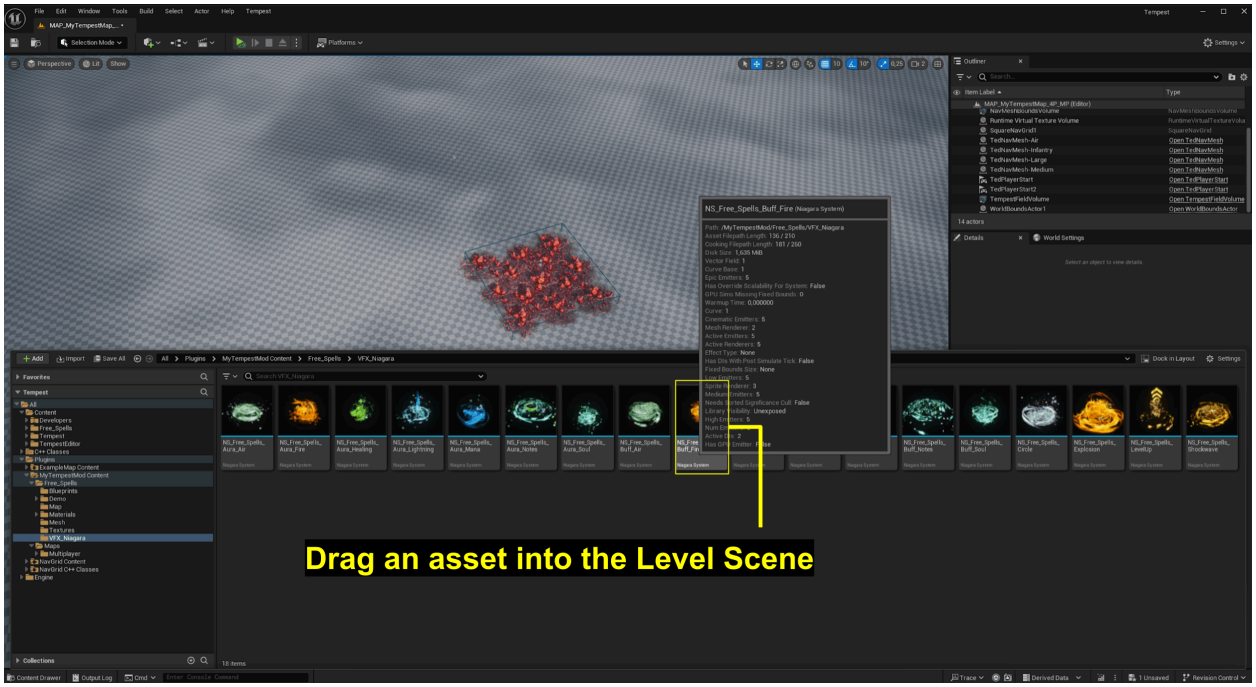
Typical path (if you moved your Map Editor project) would be:
**C:\Users\Username\Documents\Unreal Projects\Map Editor\Mods\MyTempestMod\Content**



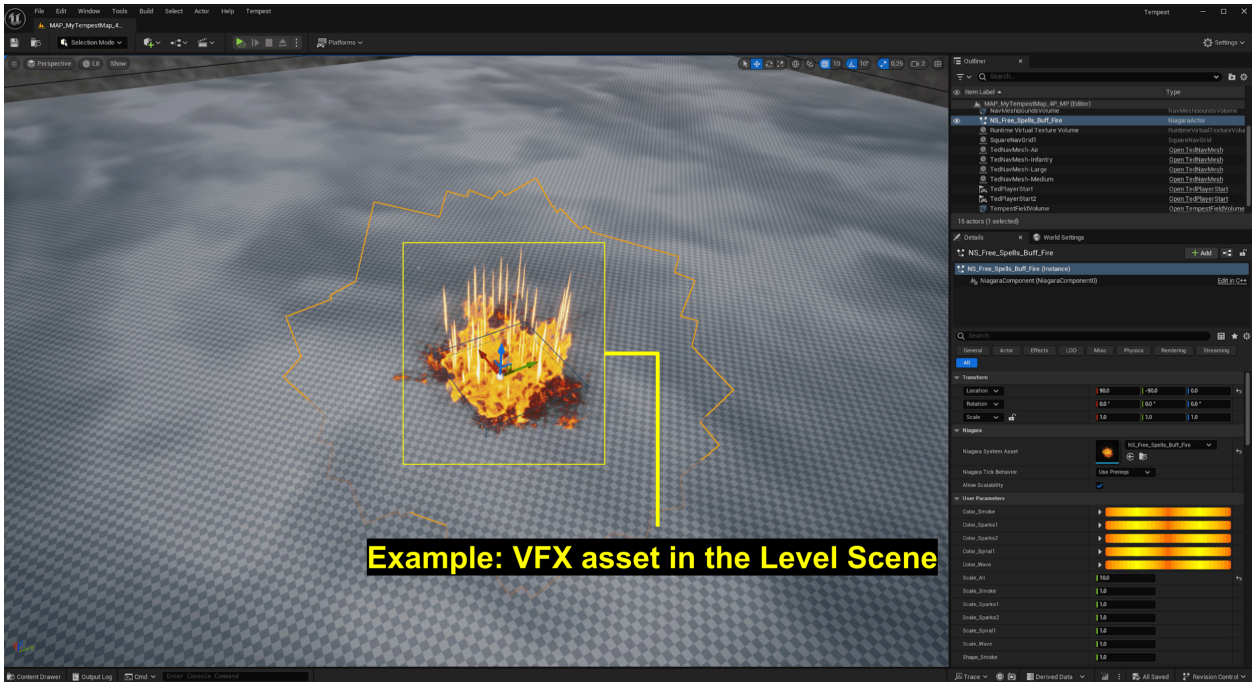

Once the migration is complete, your asset pack should now be located into the 'MyTempestMod' folder (or whatever the name of your Mod currently is):



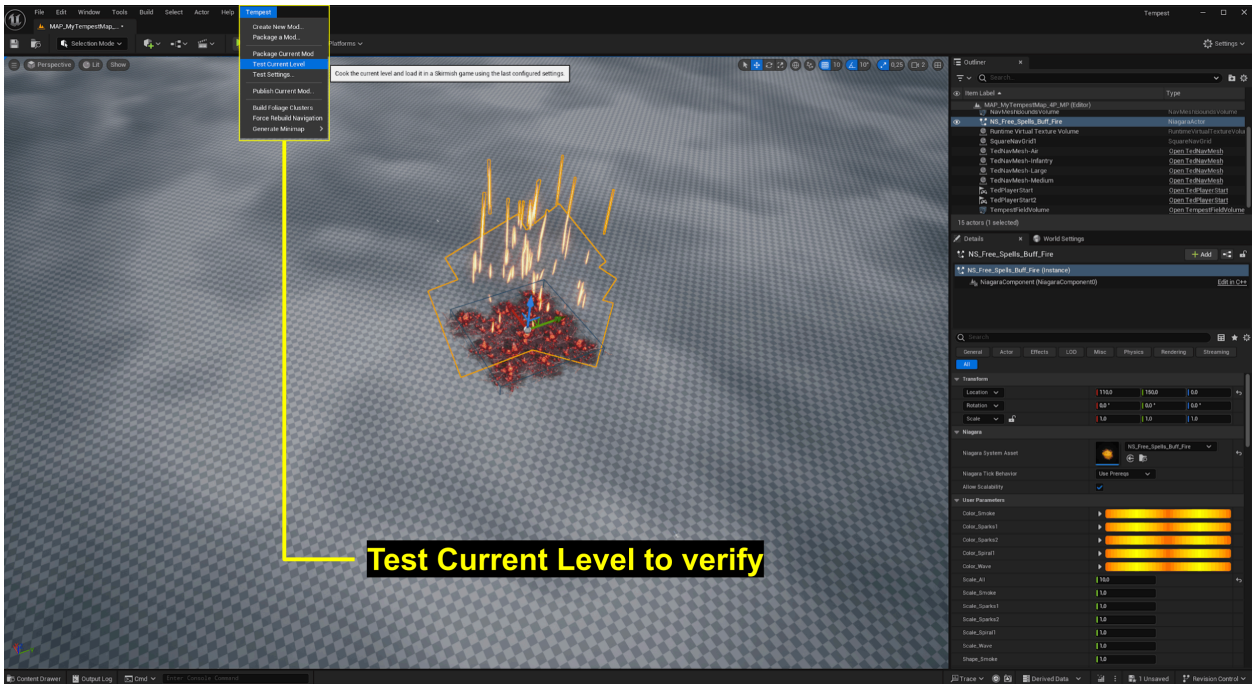

To verify that the assets are ready to use, **>DRAG<** an asset from inside the asset pack folder into the Level Scene:



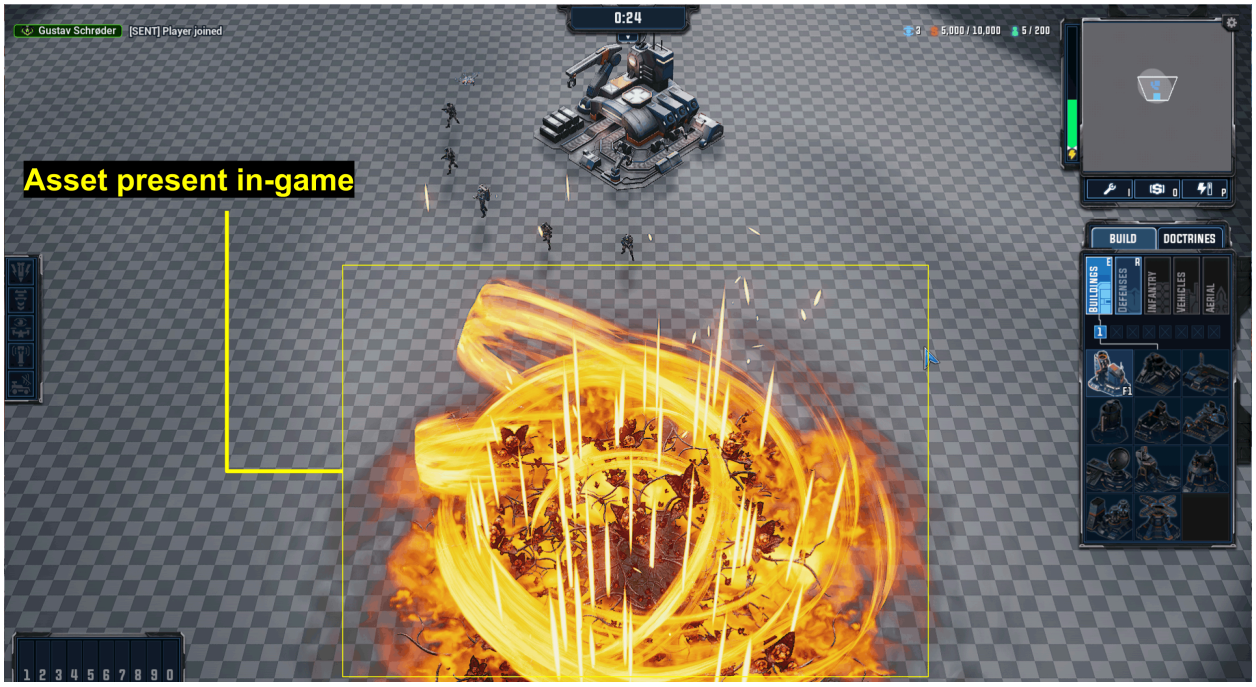

You should be able to see the asset inside your Level now (and scale it to whatever you need it to be, size-wise):




To fully verify that the asset is indeed in your Map scene, **>CLICK<** on ‘Test Current Level’ in the top tempest menu, as you have done for previous testing:




Your selected asset should now be present inside the game, when Tempest Rising launches:




**NOTE**

*If your asset still is not showing up in-game, you may have:*

1) *Migrated to the wrong content folder. If so, you need to re-do the Migration step, as previously described*

2) *Moved the assets somewhere after migrating. Unreal is finicky with dependencies, so Moving assets (instead of Migrating) may cause issues*

3) *The asset itself may be broken, and would not work in a default unreal project either. A test case would be to create an empty Unreal project, and check that the asset can be packaged from there as well*